JP 2002-506285

* NOTICES *

CLAIMS

[Claim(s)]
[Claim 1] It is a semi-conductor wafer processing system. Two or more processing chambers, A migration chamber including the transport station which is arranged at the core between two or more of said processing chambers, and transports a wafer between processing chambers, The sequence task-control equipment for controlling said wafer transfer device and reconstructing the schedule of the wafer by which the deadlock was carried out so that the timing of the sequence of migration of the wafer between processing chambers may be doubled and set in order The semi-conductor wafer processing system which it has.
[Claim 2] Said sequence task-control equipment Means for identifying the wafer by which the deadlock was carried out within the supply source processing chamber Means for ordering a wafer transport station to transport said identified wafer to a position in readiness, and to release said supply source processing chamber by it System according to claim 1 which it has further.
[Claim 3] Said means for identifying the wafer by which the deadlock was carried out Means for determining the destination chamber of a wafer and determining whether the resource is available at the time of the schedule of migration to the destination processing chamber meant from the supply source processing chamber Means for surely discriminating the wafer from the wafer by which the deadlock was carried out, when a destination processing chamber cannot be used System according to claim 2 which it has further.
[Claim 4] Said means for identifying the wafer by which the deadlock was carried out System according to claim 3 further equipped with the means for making a sequence change of the wafer processing sequence so that the schedule which transports said wafer by which the deadlock was carried out from a position in readiness to the meant destination chamber may be constructed, when the resource becomes available.
[Claim 5] The system according to claim 4 which has said position in readiness in a load lock.
[Claim 6] The system according to claim 1 said whose wafer by which the deadlock was carried out is a wafer of sequence medium.
[Claim 7] An orientation chamber is included further. System according to claim 1 said whose wafer by which the deadlock was carried out is an orientation chamber.
[Claim 8] The system according to claim 1 by which said sequence task-control equipment uses a wafer migration queue, chamber DS, wafer management data structure, and a wafer sequence list.
[Claim 9] The system according to claim 8 by which said wafer sequence list includes the information which identifies each wafer in said two or more wafers uniquely using wafer discernment data.
[Claim 10] The system according to claim 8 by which said wafer migration queue includes the information which identifies each supply source chamber and destination chamber of said wafer in said two or more wafers.
[Claim 11] The system according to claim 8 by which said chamber DS includes the information about a chamber processing parameter.
[Claim 12] The system according to claim 8 by which said wafer management data structure includes the information which identifies the supply source of each wafer, the condition of each wafer, a supply source cassette identifier, and a supply source slot identifier.
[Claim 13] It is the approach of processing a wafer by the multi chamber wafer processing system. Process which identifies the wafer by which the deadlock was carried out, Process which moves said wafer by

which the deadlock was carried out to a position in readiness until the destination chamber of said wafer by which the deadlock was carried out becomes available, Process which moves said wafer by which the deadlock was carried out to said destination chamber The approach for processing a wafer by the multi chamber wafer processing system of including.

[Claim 14] The approach according to claim 13 which has said wafer by which the deadlock was carried out in an orientation chamber.

[Claim 15] Said discernment process Process which investigates a wafer migration queue and identifies the destination chamber of a specific wafer Process which investigates chamber DS and determines whether said destination chamber is available Process which discriminates a specific wafer from a deadlock when a destination chamber cannot be used The approach according to claim 13 of including further.

[Claim 16] Process which determines the time of said destination chamber becoming available Process which updates a wafer migration queue by specific wafer discernment data in order to promote group repair of the schedule for moving a specific wafer to a destination chamber The approach according to claim 15 of including further.

[Claim 17] The approach according to claim 16 of including further the process which attaches priority to the wafer of either of said wafer migration queues, after said wafer migration queue is updated.

[Claim 18] An approach including the process maintained as the door of an orientation chamber was opened while being returned to a load lock until said wafer by which the deadlock was carried out uses an orientation chamber as a supply source chamber, sequential migration of the wafer of further plurality [approach / said] is carried out from a load lock at an orientation chamber and a destination chamber becomes available according to claim 13.

[Claim 19] The approach according to claim 18 of including further the process which moves directly the wafer by which orientation was carried out to said destination chamber from said load lock.

[Claim 20] Sequential processing is carried out within two or more processing chambers by which each wafer was adapted for one or the integrated-circuit process beyond it according to the process recipe. It is the approach of carrying out multi chamber processing of the wafer that the schedule of two or more wafers is constructed. The process which assembles the wafer sequence list which identifies the timing and sequence that said two or more wafers are processed in two or more of said processing chambers and different processes, Process which processes said wafer according to said wafer sequence list About each wafer identified by the wafer sequence list The process which determines the condition of the next destination processing chamber of the processing sequence, and the destination processing chamber at the time of completion of a current process process from the condition current [the] in a supply source processing chamber, and its process recipe, Process which transports a wafer to a destination processing chamber when the condition is available When a destination processing chamber is utilization disabling Process which transports a wafer to a standby chamber Process into which a wafer sequence list is changed about the wafer transported to the standby chamber in order to reconstruct the schedule of a wafer processing sequence How to include.

[Claim 21] The process which determines whether either of the wafers in a wafer sequence list is transportable, and when all the wafer cannot be transported, either The process which opens both load lock and an orientation chamber to a migration chamber, By transporting wafer to said orientation chamber from said load lock, carrying out orientation of said wafer within said orientation chamber, and transporting said orientation finishing wafer to said load lock Process which performs orientation of a wafer Until a destination chamber becomes available Process which repeats said wafer orientation to other wafers in said load lock Process which transports said load lock and an orientation chamber to closing, and transports a wafer to a destination chamber The approach according to claim 20 of including further.

[Translation done.]

* NOTICES *

DETAILED DESCRIPTION

[Detailed Description of the Invention]
[0001]
[Field of the Invention]
This invention relates to the multi chamber semi-conductor wafer processing system which has two or more processing chambers in more detail about an integrated-circuit wafer processing system.
[0002]
[Description of the Prior Art]
The multi chamber semi-conductor wafer processing system (known as a cluster tool) which can perform two or more processes sequential and simultaneous to works, such as a semi-conductor wafer, is shown at U.S. Pat. No. 4,951,600 (Maydan I) published by Maydan and others and U.S. Pat. No. 5,292,393 (Maydan II) similarly published by Maydan and others, and includes these disclosure in this application description by reference.
[0003]
Maydan I and Maydan The cluster tool indicated by II is an integrated vacuum processing system which contains fundamentally one or the load lock chamber beyond it, a migration chamber, and two or more vacuum processing chambers currently connected to the load lock chamber and the migration chamber through the slit which can be closed selectively [each chamber]. One arranged in a chamber or the chemical vapor deposition of the wafer beyond it, physical vacuum evaporationo, dirty processing, rapid heat annealing, etc. can fit each processing chamber so that one or integrated-circuit processing beyond it may be performed. The load lock chamber has incorporated the external wafer elevator fitted so that a wafer might be arranged at the inlet port of a load lock chamber, and the internal elevator fitted so that a wafer might be moved to the location or migration location contiguous to the outlet of a load lock chamber.

[0004]
The wafer transport station arranged in the center is attached in a migration chamber, and, generally contains a level wafer maintenance blade. A wafer transport station performs revolution of a blade, expanding, and level luffing motion, and arranges a blade to an external elevator, an internal elevator, and a processing chamber selectively.
[0005]
In the usual processing mode, process control equipment transports a wafer to the maintenance plate of an internal elevator through the inlet port of one load lock from the slot in the migration cassette of an external elevator. After a wafer is loaded into a cluster tool, the inlet port of a load lock is sealed. Subsequently, a load lock, a migration chamber, and a processing chamber are exhausted with a pump by even the vacuum level for processing. A real-time automatic process sequencer starts, the 1st wafer is moved to a processing chamber, the inlet port is closed, the processing is started, the 2nd wafer is moved to a processing chamber, the inlet port is closed, and the processing is started. The multi chamber continuation sequential process of two or more wafers is attained by planning the path of the wafer between processing chambers while a cluster tool is closed and being in a vacua by that cause. If processing of a wafer is completed, it will usually be returned to the specific plate of an internal elevator until processing of all the wafers in a specific lot is completed and is returned. Subsequently, a cluster tool is opened by atmospheric air and a wafer is taken out by the migration cassette of an external elevator.

[0006]
Adjustment and control of the chemical preparation of the element of a multi chamber processing system are offered by the real-time multitasking control program which makes possible interactive user input and system monitoring. Generally, a sequencer task module reads a wafer sequence list including the discernment and process recipe, or sequence of each wafer on an internal elevator maintenance plate, and plans the chemical preparation performed migration of the wafer between processing chambers, and there.
[0007]
If the processing chamber of the recipe sequence of a wafer becomes available, a wafer will be processed next from a degree in order of a wafer sequence list. Although this often functions theoretically, depending on the case, a wafer will be in a "deadlock" condition by the process sequence. This is usually generated, when [to which the process recipe of a specific wafer continues after short-time down stream processing] down stream processing of long duration is included all the time. Although a short-time process process is planned and is completed comparatively promptly, when migration to a processing chamber is tried in the case of a long duration process process, a wafer is interrupted by the destination processing chamber in "under an activity" or processing mode for the same wafer before that. If this occurs, both chambers (a supply source and destination) will be as a matter of fact among the time amount of the die length of a process for a long time in "a condition in use, i.e., the condition for processing after it which cannot be used," for the die length of prolonged processing. The deadlock of the wafer in the chamber of a supply source is carried out literally, and it waits for the destination chamber of a wafer to become available.
[0008]
Also in the case of a short process, this generates an orientation process etc. intrinsically again. Although an orientation process can be performed comparatively promptly, subsequently a wafer waits repeatedly for the following processing chamber in a process sequence to become available within an orientation chamber. This blocks using an orientation chamber for other regularly processible wafers, when prior orientation is carried out and the following processing chamber in a process sequence is subsequently being able to use. This prior orientation becomes economization of most time amount at the subsequent process in a process sequence later.
[0009]
Therefore, in order to reduce or avoid the situation which carried out the deadlock, the approach and equipment which reconstruct the schedule of a system resource are required technically.
[0010]
[Means for Solving the Problem]
This invention offers the multi chamber processing system equipped with the multitasking control containing the process sequencer which can identify the workpiece or wafer which foresaw and carried out the deadlock of the point of a process sequence. Once the wafer by which the deadlock was carried out is identified, it will be removed from a real-time sequence process by returning it to a load lock preferably by transporting the wafer to the position in readiness in a system. The wafer by which the deadlock was carried out is a load lock, and when the resource becomes available, it waits to be carried out [with which a schedule is reconstructed so that it may be sent to the original destination chamber].
[0011]
Therefore, since a system throughput is increased remarkably, this invention offers the approach and equipment for canceling lock out of the resource of a multi chamber processing system (that is, cluster tool) by using a pre sequencer. With one operation gestalt of this invention, the wafer of the identified dead lock status is a wafer of sequence medium which ended one of the processing process of the, and is waiting to be transported to a destination chamber for the following processing process by the supply source chamber. A destination chamber makes a system resource blockade by making the supply source chamber into the busy condition, while waiting to become available, and probably does not delay other wafers beyond one or it, and a wafer moves the wafer of sequence medium which carried out the deadlock to a position in readiness. For example, a wafer is moved to the internal elevator of a load lock. After migration, a supply source chamber becomes usable, in order to continue processing of other wafers in a system. Subsequently, this invention reconstructs the schedule of the wafer currently held in the position in readiness, when a destination chamber becomes available.
[0012]

With the 2nd operation gestalt of this invention, this invention identifies the wafer with which orientation of [in an orientation chamber] was carried out as a wafer by which the deadlock was carried out. That is, since the 1st processing chamber of the process recipe is using [be / it] it (utilization is impossible), the wafer is not processible. As compared with almost all chemical preparation, for short time amount required for an orientation process, an orientation chamber may serve as main sources of release of a deadlock wafer, and all the throughputs of a system may be reduced. I hear that almost all wafers must pass an orientation chamber, and another reason an orientation chamber becomes a problem has them, and if this does not use an orientation chamber more effectively, it becomes the cause which causes delay. This invention solves this problem by discriminating the wafer with which orientation of [in the orientation chamber which that following destination chamber is using] was carried out from a deadlock wafer, and returning the wafer by which orientation was carried out to a load lock or the convenient wafer position in readiness of a certain others. Thereby, a system continues using an orientation chamber more effectively and it becomes possible to increase a throughput.

[0013]

With the 3rd operation gestalt of this invention, when all processing chambers are using it, this invention carries out orientation of the wafer in the load lock which is waiting to be processed first, and then offers preferably a position in readiness and the prior orientation process returned to a location at those beginning in a load lock. therefore, since the sequence is changed in order to raise the effectiveness of a system even if the orientation process of a specific wafer is performed where of the usual process sequence, in order to save the processing time, it can come out and perform excluding it.

[0014]

With the operation gestalt which this invention illustrated, these approaches can be performed by the process control sequencer which controls the cluster tool containing the load lock chamber, the migration chamber, the buffer chamber, the orientation chamber, and two or more vacuum processing chambers beyond one or it, for example, the programmed general purpose computer. Each processing chamber is adapted so that one or integrated-circuit processing beyond it (recipe) on a wafer may be performed. The wafer transport station arranged at the core of a migration chamber is used in a wafer in order to move between various chambers, so that a multistage story processing sequence may be promoted.

[0015]

[Embodiment of the Invention]

If the following detailed description is combined with an attached drawing and read, these of this invention and the other objects, a mode, and the description will be understood more clearly, and will more often be explained.

[0016]

In order to promote an understanding, the same element common to a drawing is shown using a same reference figure, when possible.

[0017]

Drawing 1 is the top view of the suitable operation gestalt of the multi chamber semi-conductor wafer processing system 10 (cluster tool) which operates according to this invention. A cluster tool is fitted so that vacuum processing of the workpieces, such as a silicon wafer for the integrated circuits of a very-large-scale-integration (VLSI) mold, may be carried out especially. The cluster tool 10 equips the sealed general target which has eight side attachment walls 13 which demarcate the vacuum migration enclosure or a chamber 14 with the main frame or housing 12 of an octagon.

[0018]

The cluster tool 10 contains four processing chambers (PC1-PC4) 16, 18, 20, and 22, the migration chamber 14, the buffer chamber 28, the wafer orientation chamber / degasifying chamber 30, and one pair of load lock chambers 24 and 26. Each processing chamber expresses the phase or phase from which semi-conductor wafer processing differs. In this disclosure, it is considered that the buffer chamber 28, and the wafer orientation chamber / degasifying chamber 30 are the processing chambers of a special class. Therefore, a chamber is only the vocabulary "a processing chamber" or a thing which includes the chamber of all the forms in an accessible cluster tool according to a transport station.

[0019]

In order to realize wafer migration between these chambers, the migration chamber 14 includes the 1st robot type transport station 82, for example, a single blade robot, (SER). Generally, a wafer 15 is put into

the migration cassette 25 made from plastics of a load lock chamber arranged in 24 or 26 one, and is carried in a system from a storage area. The robot type transport station 82 transports every one wafer 15 to a wafer orientation chamber / degasifying chamber 30 from a cassette 25 at once. Generally, the buffer chamber 28 is not used until a wafer is processed within one or the processing chambers 16, 18, and 20 beyond it, and 22. Each wafer is carried and carried by the wafer migration blade 86 arranged in the far edge 84 of the robot device 82. It expands and contracts so that it may be expressed by the arrow head 88, and although the condition of having contracted is shown, this device is rotated so that it may be expressed by the arrow head 90. The orientation of migration is controlled by the control unit 70.
[0020]
A control unit 70 controls processing and wafer migration which are performed by the cluster tool 10. A control device includes exchange circuits, such as memory for storing a microprocessor (CPU) and a control routine and a power unit, a clock circuit, and cache **. A control device 70 also contains I/O peripheral devices, such as a keyboard, a mouse, and a display, again. A control device 70 is a general purpose computer which performs sequencing and scheduling which promote processing and migration of a wafer. The software routine which controls a cluster tool is stored in memory, and in order to promote control of a cluster tool, it is performed by the microprocessor.
[0021]
A control device 70 directs a transport station according to a process schedule, and arranges a wafer to the processing chambers 16, 18, 20, and 22. In order to promote migration of such a wafer, the migration chamber 14 is enclosed by four processing chambers 16, 18, 20, and 22, and can access them. If processing is completed within a processing chamber, a transport station 82 will move the completed wafer from a processing chamber, and will transport a wafer to the buffer chamber 28. A wafer is taken out from a buffer chamber next and arranged at the load lock chambers 24 or 26.
[0022]
Drawing 2 shows the block diagram of the control device 70 which performs the system process control software for cluster tool 10, and generates automatic control and process sequencing. A control device 70 can be realized in more detail as a general purpose computer (for example, a mainframe computer, a workstation, a personal computer, or a microcomputer) for controlling a multi chamber processing system. A general purpose computer can be equipped with a central processing unit (CPU) or PUROSSESSA 72, memory 71, ROM73 and various I/O devices 74, for example, a monitor, a keyboard, and/or various storage.

[0023]
With a suitable operation gestalt, a control unit is a microcomputer, and it incorporates new system-software application so that it may mention later. The system software is expressed by the software application or module beyond one or it loaded to memory 71 from I/O device 74, for example, the MAG, or an optical disk drive, a diskette, or a tape. The system software is realizable in alternative as firmware stored in read-only memory (ROM) 73 and a prototype. As such a thing, the cluster tool control software of this invention is storable in the medium in which one or the computer read beyond it is possible. Finally, once it loads a software application, a processor 72 will perform this new software in memory, and will realize a cluster tool sequencer. Generally, a processor 72 can be used as the 680x0 mold manufactured, one, for example, Motorola, of the microprocessor of marketing of arbitration.
[0024]
It plans realizable as a circuit device in which some of process processes discussed as a software process here collaborate with a microprocessor in hardware, and various process processes are performed. the process realized by software although the control device is drawn as a general purpose computer programmed to perform various scheduling routines -- application -- it is realizable as hardware, such as a specific integrated circuit (ASIC) or a discrete circuit component. As such a thing, the process process indicated here has the intention of what is interpreted broadly as what is equally performed with the combination of software, hardware, or those arbitration.
[0025]
Sequencer software is multitasking and is divided into the real-time task orientation module called with interruption system from the timer which generates a periodical alerting signal. As shown in drawing 2 , basic sequencer software contains the sequencer task module 200, the screen task module 202, the low-speed task module 204, the chamber task module 206, and the buffer chamber task module 208. When

called, each modules 200-208 complete interruption and those processings to processing of a low-speed module more, and return at the exit point of the last of the routine subsequently interrupted. By this conventional multitasking hierarchy actuation, more important actuation can be processed in the real time, without having serious effect for processing on other tasks in a system. When other modules publish the command with which it performs during processing of the module ordered later changing the DS to be used or during the processing to another module by leaving other messages or data for modules respectively to the outlet, as for the above-mentioned modules 200-208, which of other modules can communicate.

[0026]

Generally, the sequencer task module 200 controls sequencing of wafer manufacture or the process for every process including a process recipe program and a washing recipe program. A module 200 operates based on the DS which indicated the sequence and chemical preparation of a process of the wafers loaded into load locks 24 and 26, those processes, and a washing recipe, i.e., various chambers, i.e., a wafer sequence list, (WOL). A process and a washing recipe are respectively related with the specific wafer of WOL through another DS (WMDS), i.e., wafer management data structure, by the pointer to information. The sequencer task module 200 which operates based on the information on WOL and WMS publishes a command to a chamber task module so that the process process of the specific process recipe for a specific chamber or a washing recipe may be performed to right time amount. A wafer moves various chambers with the sequencer task module 200 which operates based on the DS which indicates that migration actuation of a system agrees in the chemical preparation of a specific chamber, i.e., a wafer migration queue, (WMQ). When a wafer is in WMQ, a sequencer module is ordered to the buffer task module 208 to perform migration by control of the entrance slit bulb of a chamber, and the wafer transport station 82.

[0027]

The chamber task module 206 deals with various recipes and the washing recipes for the integrated-circuit process on a wafer including starting of actual control of a vacuum valve, various ionization processes of a chemical, and RF power source etc. The chamber task module 206 supervises and controls the process which happens by the specialized processing chamber by the actuation based on DS (CDS), i.e., chamber DS.

[0028]

The screen task module 202 makes it possible to leave a message or a command during those activation, before it programs the data of specification [an operator] in the DS of other modules and they are performed by the task module (or other program objects) by the interactive approach. One of the bodies of the screen task module 202 is an interactive text editor, and he is enabled to input data and a command, or for an operator to edit through an operator interface or a display, and to store them in the control for a process, and an administrative module (not shown). The screen task module 202 makes it possible to display a condition with various operators, an alarm, and surveillance intelligence during activation of the automatic-control sequence of a system 10 again.

[0029]

The low-speed task module 204 is also formed in order to supervise the specific task of the system performed comparatively at a low speed and to take timing. For example, the RF coil of a specific chamber was energized or the washing time amount of a chamber can be supervised by determining the amount of the active time amount.

[0030]

Here, the DS used by process control is explained more to a detail in relation to 6 from drawing 3 . WOL DS is indicated by the detail by drawing 3 . Wafer sequence lists are two or more information blocks 1 and 2... It has n and one of the wafers by which the current plan is carried out so that it may be processed into each by the wafer processing system 10 is indicated. Each information block 1, for example, the block displayed as 210, contains the lot number to which the wafer identification number and it which identify a wafer uniquely belong. the wafer management blocks 1 and 2 corresponding to [an information block 210 controls processing of a wafer further, and] the information block 210 of a wafer sequence list, and ... the pointer to WMD (drawing 6) which has n is included further. There are the two flag fields behind these fields. A thing for one to identify a wafer as a process wafer by which the deadlock was carried out, and another are for a deadlock being carried out and identifying it as a wafer (prior orientation) by which orientation was carried out. The field of the last of an information block 210 is a group chamber mask

which identifies the group of a processing chamber who can construct the processing schedule of a wafer. The remaining information blocks 2 and 3 of WOL ... n is constituted like an information block 210 and includes the information about the remaining wafer processed with a cluster tool.
[0031]
Wafer management data structure is indicated in detail by drawing 6 , and is structurally similar to the wafer sequence list. WMD -- the wafer management blocks 1 and 2 and ... it has n, and each corresponds to each WOL information block, therefore it corresponds to a specific wafer. The field of the wafer management block 1, for example, the block displayed as 216, contains the lot number and wafer identification number of a wafer. In addition to it, a Status field indicates a process condition. Furthermore, there is program sequence discernment which indicates the washing recipe matched with the process recipe of a wafer and it. The field of the last of the wafer management block 216 records the cassette number and slot of the supply source with which a wafer starts the process. Ideally, such a supply source slot is a slot to which a wafer is returned after processing. The remaining wafer management blocks 2 and 3 of WMDS ... n is constituted like the wafer management block 216, and includes the information about the remaining wafer in a system 10.
[0032]
WMQ DS is indicated in detail by drawing 4 . the identifier blocks 1 and 2 of a large number including the field which identifies the wafer which a wafer migration queue is a list of wafers which need migration, and should move, and its moving trucking (a supply source and destination), and ... it consists of n. The identifier block 1 displayed as 212 has one and the supply source for memorizing the pointer to the wafer information block in the wafer sequence list which identifies the wafer which should move and a destination processing chamber, or the field including two another fields for identifying the slot of an elevator or a cassette, when a supply source or a destination is a cassette or a load lock. The remaining identifier blocks 2 and 3 of WMQ DS ... n is constituted like the identifier block 212 and contains the same data of the wafer of an and also [migration is the need].
[0033]
Chamber DS is indicated in detail by drawing 5 . Chamber DS is a list of processing chambers and those conditions. chamber DS -- much condition blocks 1 and 2 and ... consisting of n, the condition block 1 displayed as 214 includes the field which identifies the remaining time amount of the recipe process under activation by the chamber in the figure of a meaning, and its chamber. Another field is used as a flag in which it is shown [under an activity of a chamber or] whether it is vacant, and when set, it shows that a chamber is using it and the process recipe process or the washing recipe process is performed. Since the specific chamber has processed the wafer of the number only of which since washing of the last or the number of wafers is stored, the last field is used. The remaining condition blocks 2 and 3 of processing chamber DS ... n is constituted like the condition block 214 and contains the same data about other chambers in a system.
[0034]
Here, in relation to drawing 7 , the sequencer task module 200 containing the pre sequencer constituted according to this invention is explained in more detail. The main program 218 of a sequencer task module is a list of routines performed every 50ms. Generally, the task module 200 is block 220, processes the incoming message from other routines, data, or a command, and starts it by updating the information from other routines which is needed in order to perform the following task.
[0035]
With block 222, **** of a process, and in order to mainly reduce or remove the deadlock of a wafer, a module 200 foresees the point, constructs the migration schedule of the wafer between 30 from the above-mentioned processing chamber 16, and performs the dynamic read-ahead routine or pre sequencer which determines priority. It becomes possible for a system 10 to use a chamber more effectively and to raise the throughput of a cluster tool by this prior sequencing.
[0036]
The dynamic read-ahead program 222 investigates the wafer sequence list used in order that the sequencer task module 200 may set the sequence of processing in order, and determines which wafer within a sequence will cause the problem on a throughput in the future. A program changes the sequence of the wafer discernment data in a wafer sequence list, and a wafer migration queue, and eases or removes these problems from the list of problem wafers with which it was identified within the sequence. As for

migration of the wafer for reducing these problems, the correction to be migration before migration of which type of wafer has priority determined by whether a throughput is improved most and performs processing after it is made. Since the pre sequencer routine 222 is performed every 50ms, once a problem is identified, it will be corrected in an instant.
[0037]
After it publishes a command to the buffer chamber task module expressed by block 224 and a wafer moves to those suitable chambers, a command is published by the chamber task module of block 226 so that either the process recipe of a wafer or the washing recipe of a specific chamber may be set in order, so that the sequencer task module 200 after WMQ and WOL, or the required information on other were updated may move a wafer by the sequence beforehand defined as demarcated by WMQ. As such a thing, a chamber task starts with block 228. Once a chamber task starts, the task sequencer module 200 will be dormant with block 229, and will wait for the following alarm clock call.
[0038]
The more detailed functional flowchart of the dynamic read-ahead routine 222 is indicated from drawing 8 a to 8g. It is the general function of the pre sequencer program expressed with block 222 being explained by drawing 8 a in full detail, and the 1st function of the block 230 there foreseeing the point of a wafer sequence list, and looking for a sequence medium deadlock wafer. Although these are ready for completing a part of those processings and moving to the following processing chamber, they are the wafers with which it became clear that the following processing chamber was using it. However, even if they stagnate a system by not opening the processing chamber or resource with which current arrangement of them is carried out again, they are. This produces the possibility of a deadlock stopped until this and the substantial part of the possible floor to floor time of other wafers become as [receive / the destination chamber of the wafer / a deadlock wafer] (available). The solution of this problem is returning it to the position in readiness of a system 10 preferably at a load lock chamber until it takes out the wafer which carried out the deadlock from that supply source processing chamber and the destination chamber of that wafer becomes available. Thereby, while the deadlock wafer is waiting for the destination chamber, it becomes possible to process other wafers by the opened supply source chamber. A load lock is the example of a position in readiness. A deadlock wafer can be temporarily arranged to the process chamber which is not used or the special position-in-readiness chamber which can be used. A position-in-readiness chamber can include the wafer elevator for storing two or more deadlock wafers which are waiting for the processing chamber.
[0039]
The next function of block 232 is investigating whether there being any wafer [finishing / orientation] which foresaw the point of a wafer sequence list and carried out the deadlock into the orientation chamber 30. Although these are ready for completing orientation and moving to the following processing chamber, they are the wafers with which it became clear that the following processing chamber was using it. However, by not opening for other wafers' processing of the orientation chamber which contains the orientation finishing wafer which carried out the deadlock again, even if they stagnate a cluster tool, they are. This produces the possibility of a deadlock which many of possible processings of the wafer of this and others stop until it comes to be able to perform preparation whose destination chamber receives an orientation finishing wafer. The solution of this problem is the position in readiness of a system 10, and returning to a load lock preferably about it until they take out again the wafer which carried out the deadlock from the orientation chamber 30 and can use a destination chamber. While the orientation finishing wafer which carried out the deadlock is waiting for migration to a destination chamber by this, it becomes possible to carry out orientation of other wafers by the opened orientation chamber. The following two functions in blocks 234 and 236 are the same. They foresee the point of WOL, investigate whether there are any process wafer and returned orientation finishing wafer returned to the position in readiness or the load lock, and reconstruct those schedules for future processing.
[0040]
Once it is attained discernment of a problem wafer (what is easy to become deadlock voice), and that a schedule reconstructs (a) Return the wafer which carried out the deadlock to a load lock, and other wafers are moved to the processing chamber or orientation chamber by which (b) release was carried out. And motion of the wafer transport station 82 for moving the wafer with which the schedule of others containing the load lock wafer which reconstructed the (c) schedule was carried out can attach priority by examining

the class of wafer with block 238. The class of wafer contains the wafer which carried out the deadlock, the wafer which carried out orientation, the wafer of sequence medium, and a prototype. A trial shows clearly what kind of wafer is listed by the wafer migration queue, and, subsequently is reorganized based on the class of wafer which wafer discernment data should move. For example, although the wafer of sequence medium which carried out the deadlock is orientation ending, it has priority higher than a raw wafer. Furthermore, although it is orientation ending, a raw wafer has priority higher than the wafer which must still be picked out from a load lock, although orientation is not carried out. After attaching priority to a wafer migration queue, the wafer within a queue moves to each of those destination as it is shown in block 240.

[0041]

From now on, the detail routine which realizes the block 230 of drawing 8 a in order to identify the wafer of sequence medium in which the deadlock was carried out by the identification read ahead is explained in more detail in relation to drawing 8 b. Although all the wafers in a wafer sequence list are inspected by the routine of drawing 8 b before they return to main routine, they indicate the process of only one wafer for **** to a term ** sake. It is determined whether a routine has the destination chamber of a large number which can read chamber discernment of the chamber (supply source chamber) containing the present wafer first with block 242, and can receive a wafer at the process of whether it is a group chamber and a degree that is,. If a chamber is not a group chamber, a routine will use the found specific chamber ID number, and will progress to block 246. When chamber discernment shows that the wafer is contained in a group chamber, a chamber available next is chosen in a group by reading the remaining time amount of a recipe with block 244 by investigating the chamber DS of each chamber of the group. A program continues discernment of a supply source chamber, and discernment of a destination chamber within block 246, and it is determined whether a destination chamber is available to the next migration. The availability of a destination chamber is examined by the busy status flag of chamber DS, and the residual time of the recipe field.

[0042]

When a chamber is available, wafer discernment data are inputted into wafer migration queue DS with block 252, and if it is the entry of the last of the wafer sequence list which this should investigate, a program will return. When a chamber cannot be used, with block 250, a deadlock flag is set to the specific wafer, and discernment of the following processing chamber which is under activity now is saved on a wafer sequence list. Subsequently to a load lock the destination chamber field of the wafer is changed, and wafer discernment data are inputted into a wafer migration queue with block 252. As such a thing, a wafer moves to a load lock after activation of a buffer chamber task module. If all the wafers of WOL are examined, a routine will return to a main task module after that.

[0043]

The detailed routine which realizes the block 232 of drawing 8 a in order to identify the orientation finishing wafer in which the deadlock was carried out by the dynamic read ahead is explained in more detail in relation to drawing 8 c. Although all the wafers of a wafer sequence list are inspected by the routine of drawing 8 c before they return to main routine, they indicate the process of only one wafer for **** to a term ** sake. It is determined whether a routine has the destination chamber of a large number which can read chamber discernment of the chamber (supply source chamber) containing the present wafer first with block 254, and can receive a wafer at the process of whether it is a group chamber and a degree that is,. If a chamber is not a group chamber, a routine will use the found specific chamber ID number, and will progress to block 258. When chamber discernment shows that the wafer is contained in a group chamber, a chamber available next is chosen in a group by reading the remaining time amount of a recipe with block 256 by investigating the chamber DS of each chamber of the group. A program continues discernment of a supply source chamber, and discernment of a destination chamber within block 258, and a destination chamber determines whether it is available as the migration to which the schedule of the degree was carried out. The availability of a destination chamber is examined by the busy status flag and the recipe residual time field of chamber DS.

[0044]

It examines with block 260, when the following chamber is available, wafer discernment data are inputted into wafer migration queue DS with block 264, and if it is the entry of the last of the wafer sequence list which this should investigate, a program will return. When a chamber cannot be used, with block 262, a prior

orientation flag is set to the specific wafer, and discernment of the following processing chamber which is under activity now is saved on a wafer sequence list. Subsequently to a load lock the destination chamber field of the wafer is changed, and wafer discernment data are inputted into a wafer migration queue with block 264. As such a thing, a wafer moves to a load lock after activation of a buffer chamber task module. If all the wafers of WOL are examined, a routine will return to the block 232 of drawing 8 a after that.
[0045]
After processing a sequence medium wafer and an orientation finishing wafer, it moves from a routine to drawing 8 d, and a routine does not have the returned process wafer and the returned orientation finishing wafer, or inspects a wafer sequence list, and reconstructs those schedules towards the original destination chamber here. The routine of drawing 8 d follows and realizes the process of the approach expressed by the blocks 234 and 236 of drawing 8 a. About all the wafers in a wafer sequence list, a routine performs the group of steps 266-278, and starts him by examining whether the information block by which is block 266 and current addressing is carried out by the wafer sequence list identifies the 1st wafer. Next, a routine opts for discernment of the chamber in which the wafer by which current addressing is carried out is located with block 268. Chamber ID is examined with block 270 next, and it is determined whether it is a load lock chamber. When that is not right, a program moves to the following wafer by moving to the trial of block 278. However, it may be the returned wafer when a wafer is in a load lock. A sequence medium flag is examined with block 272, an orientation finishing flag is examined with block 274, and it is determined whether either is set or not. When one of flags is set, this is a returned wafer which is waiting for a schedule to reconstruct. Moving from the following process to drawing 8 e, this realizes the schedule **** repair routine called by block 276. When it is shown by the trial of blocks 272 and 274 that the deadlock flag is not set, as for a program, the next information block of a wafer sequence list is inspected by blocks 278 and 280 return and there. After all the wafers of a wafer sequence list are inspected, a program returns to the block 234 of drawing 8 a, or 236 then.
[0046]
Drawing 8 e shows the flow chart of the routine which realizes the wafer schedule **** repair process of the block 276 of drawing 8 d. It is determined whether a routine has the destination chamber of a large number which can read chamber discernment of the chamber (supply source chamber) containing the present wafer first with block 282, and can receive a wafer at the step of whether it is a group chamber and a degree that is,. The identification information of a group chamber is contained in a group mask. If a chamber is not a group chamber, a routine will use the found specific chamber ID number, and will progress to block 286. When chamber discernment shows that the wafer is contained in a group chamber, a chamber available next is chosen in a group by reading the residual time of a recipe with block 284 by investigating the chamber group mask and the chamber DS of each chamber (that is, when the close chamber is in a group mask). With block 286, a routine inspects chamber DS and it is determined whether a destination chamber (target) is available. It refers for whether the target chamber of a routine is available with block 288. When enquiry is answered in the negative, a routine returns to the block 276 of drawing 8 d. Therefore, a wafer does not rehave a schedule constructed but it must wait for it so long. When enquiry of block 288 is answered in suitable, a routine goes into block 290 and updates WMQ DS. As such a thing, a wafer moves to the chamber for processing from the position in readiness of a load lock after activation of the following buffer chamber task. Next, a routine returns to the block 276 of drawing 8 e.
[0047]
After identifying the sequence intermediate-processing-intermediate-treatment wafer by which the deadlock was carried out, and the orientation finishing wafer by which the deadlock was carried out, reconstructing the schedule of the migration, and identifying the returned sequence medium wafer and the wafer by which prior orientation was carried out and reconstructing the schedule of the migration, a program determines the priority of a wafer migration queue. The functional flowchart where the step of this process is detailed is described in more detail to drawing 8 f which realizes the block 238 of drawing 8 a. A wafer migration queue begins a process and a washing recipe, a pre sequencer, a screen task module (operator control), and a prototype, and is loaded in random sequence from various programs with the need of moving a wafer. The routine of drawing 8 f redoes sequencing of these migration, attaches priority to **** which may be generated in a process, and prevents the lock out. Sequencing of a wafer sequence list is redone and the raw wafer in a load lock is put on a wafer [finishing / the sequence medium when the sequence medium and orientation finishing wafer by which was set to the second in the washing wafer, and

the deadlock was carried out in the first place / of priority / to the third in the processed wafer was returned in the load lock by the fourth, and orientation], and the fifth. Since the processing chamber which needs washing is not stopped by this approach, that chamber becomes available by the time amount in which the shortest is possible. Furthermore, the completed process wafer can be picked out from a system, or can be moved to the following step, and a resource can be released by it for other wafers which should move by the second priority. A wafer [finishing / the sequence medium by which the deadlock was carried out, or orientation] moves by the third priority, and releases a specific processing chamber or a specific orientation chamber, respectively. Subsequently, the returned orientation finishing wafer and the returned sequence medium wafer move using the interval of a schedule which may happen, and completes processing in these wafers. Finally, the raw wafer in a load lock is the tail end of a migration queue. Thereby, a new wafer does not create a new schedule until other wafers processed selectively are processed. The minimum migration of a wafer is planned by this approach, and through processing effectiveness improves remarkably.

[0048]

As a matter of fact, the program of drawing 8 g generates a command to a buffer chamber task module so that a wafer may be moved, and it realizes the block 240 of drawing 8 a as such a thing. After redoing sequencing of WMQ, migration of the wafer of a list is started. Sequential addressing of each entry of a wafer migration queue is carried out with block 302, and a command is taken out by the buffer task module so that the specific wafer for [current] an interest may be moved. a ******** [that a routine is block 304 and it is a wafer / finishing / the sequence medium to which the wafer was returned by investigating wafer discernment data with the flag of a wafer sequence list, or orientation / after the wafer migration command to the wafer of a list is taken out] -- and it is determined whether the supply source chamber of the wafer is a load lock. When both conditions are fulfilled, it was determined as what kind of wafer, or (that is, is a wafer the wafer by which the deadlock was carried out, and a wafer by which prior orientation was carried out also to it?) a prior orientation flag or a deadlock flag is reset by the wafer sequence list with block 306. After clearing these flags, a program is block 308 and it is determined whether this is the wafer of the last of a wafer migration queue. If it is not the last wafer, a program will publish return and another wafer migration command to block 302. After the last wafer command is published, a program returns to the block 240 of drawing 8 a.

[0049]

The buffer chamber task module 208 explained further in full detail in the functional flowchart of drawing 9 a to 9b is a program which controls the wafer transport station 80 and performs migration of the wafer between a processing chamber and a load lock. The buffer chamber task module 208 is called with interruption system every 20ms, and performs this actuation together with the task of others relevant to a buffer chamber. When it is determined that the buffer chamber task module 208 will have the command which moves a specific wafer with block 310, it calls the wafer migration program 312, in order to attain the task. As the above-mentioned [the wafer migration program 312], a transport station 82 lifts a wafer from the supply source chamber, and controls moving it to the destination chamber. This program controls closing motion of the entrance slit valve of a chamber again, makes migration possible if needed, and attains the separation under processing and washing. After a program receives a wafer migration command according to another description of a pre sequencer, it examines whether a destination is a slot in a cassette. When this trial is truth, a pre sequencer foresees the point of a wafer migration queue, determines whether there is any another wafer connected to the load lock, and makes the slit valve of a load lock as [open beam] after exchange of a wafer with block 316. Two or more wafer exchange can be attained as such a thing, without opening and closing the slit valve of a load lock at every exchange. Then, after a routine progresses to block 317 and completes other tasks there, it returns at the interruption exit point.

[0050]

The detail chart of the routine of drawing 9 b realizes step 316 of drawing 9 a. First, it is block 320, a program determines whether a wafer is in a wafer migration queue more, and if there is nothing, it will return a negative indicator to the wafer migration program 312. If it is, the following wafer will be examined with blocks 322 and 324, and it will be investigated whether the supply source cassette discernment is equal to the wafer cassette discernment for which it comes on the contrary. When a trial is answered in suitable, a routine determines the current location of the following wafer there following block 326. After

determining the current location of a wafer, it examines whether a program is equal to the cassette discernment for which it is [discernment] block 328 and the location comes on the contrary. In the case of yes, a program returns an affirmative response to a wafer migration program. When that is not right, it investigates whether a program relapses into block 320, examines a wafer migration queue there, and has a wafer more. The flag in which it is shown that the wafer by which the schedule was carried out is in a wafer migration list is generated by the wafer migration program so that it may be arranged at the load lock and may take out from a load lock by this, and the schedule of the current wafer is carried out so that it may move to a load lock. A slit valve door must be opened in order that close may move the wafer to which it comes to a load lock, and it is intentionally made as [open beam] so that there may be no need of closing only in order to transport after that the wear which a wafer migration program leaves and to open again later. Since there is the most amount of wafer migration in a load lock, with prior sequencing, especially this refined time amount economization raises the effectiveness of a load lock mechanism, and falls wear.

[0051]

Although this invention was explained in relation to the suitable operation gestalt, it does not mean restricting this description to the specific gestalt which specified the range of this invention, and has the intention of covering all the alternative examples included in the pneuma of this invention demarcated by the claim reversely [that], and within the limits, a modification, and an equal object.

[Brief Description of the Drawings]

[Drawing 1]

It is the outline top view of the multi chamber processing system which operates according to this invention.

[Drawing 2]

It is the pictorial system flow chart of the process control program of the control unit shown in drawing 1 .

[Drawing 3]

It is the tabular format expression of the wafer sequence list data structure used by the process control program shown in drawing 2 .

[Drawing 4]

It is the tabular format expression of the wafer migration queue DS used by the process control program shown in drawing 2 .

[Drawing 5]

It is the tabular format expression of the chamber DS used by the process control program shown in drawing 2 .

[Drawing 6]

It is the tabular format expression of the wafer management data structure used by the process control program shown in drawing 2 .

[Drawing 7]

It is the detail chart of the sequencer task module of the process control program shown in drawing 2 .

[Drawing 8 a]

It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .

[Drawing 8 b]

It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .

[Drawing 8 c]

It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .

[Drawing 8 d]

It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .

[Drawing 8 e]

It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .

[Drawing 8 f]

It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .

[Drawing 8 g]

It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .

[Drawing 9 a]

It is the detail chart of the buffer chamber task module of the process control program shown in drawing 2 .

[Drawing 9 b]
It is the detail chart of the buffer chamber task module of the process control program shown in drawing 2 .

[Translation done.]

* NOTICES *

TECHNICAL FIELD

[Field of the Invention]
This invention relates to the multi chamber semi-conductor wafer processing system which has two or more processing chambers in more detail about an integrated-circuit wafer processing system.
[0002]

[Translation done.]

PRIOR ART

[Description of the Prior Art]
The multi chamber semi-conductor wafer processing system (known as a cluster tool) which can perform two or more processes sequential and simultaneous to works, such as a semi-conductor wafer, is shown at U.S. Pat. No. 4,951,600 (Maydan I) published by Maydan and others and U.S. Pat. No. 5,292,393 (Maydan II) similarly published by Maydan and others, and includes these disclosure in this application description by reference.
[0003]
Maydan I and Maydan The cluster tool indicated by II is an integrated vacuum processing system which contains fundamentally one or the load lock chamber beyond it, a migration chamber, and two or more vacuum processing chambers currently connected to the load lock chamber and the migration chamber through the slit which can be closed selectively [each chamber]. One arranged in a chamber or the chemical vapor deposition of the wafer beyond it, physical vacuum evaporationo, dirty processing, rapid heat annealing, etc. can fit each processing chamber so that one or integrated-circuit processing beyond it may be performed. The load lock chamber has incorporated the external wafer elevator fitted so that a wafer might be arranged at the inlet port of a load lock chamber, and the internal elevator fitted so that a wafer might be moved to the location or migration location contiguous to the outlet of a load lock chamber.

[0004]
The wafer transport station arranged in the center is attached in a migration chamber, and, generally contains a level wafer maintenance blade. A wafer transport station performs revolution of a blade, expanding, and level luffing motion, and arranges a blade to an external elevator, an internal elevator, and a processing chamber selectively.
[0005]
In the usual processing mode, process control equipment transports a wafer to the maintenance plate of an internal elevator through the inlet port of one load lock from the slot in the migration cassette of an external elevator. After a wafer is loaded into a cluster tool, the inlet port of a load lock is sealed. Subsequently, a load lock, a migration chamber, and a processing chamber are exhausted with a pump by even the vacuum level for processing. A real-time automatic process sequencer starts, the 1st wafer is moved to a processing chamber, the inlet port is closed, the processing is started, the 2nd wafer is moved to a processing chamber, the inlet port is closed, and the processing is started. The multi chamber continuation sequential process of two or more wafers is attained by planning the path of the wafer between processing chambers while a cluster tool is closed and being in a vacua by that cause. If processing of a wafer is completed, it will usually be returned to the specific plate of an internal elevator until processing of all the wafers in a specific lot is completed and is returned. Subsequently, a cluster tool is opened by atmospheric air and a wafer is taken out by the migration cassette of an external elevator.
[0006]
Adjustment and control of the chemical preparation of the element of a multi chamber processing system are offered by the real-time multitasking control program which makes possible interactive user input and system monitoring. Generally, a sequencer task module reads a wafer sequence list including the discernment and process recipe, or sequence of each wafer on an internal elevator maintenance plate, and plans the chemical preparation performed migration of the wafer between processing chambers, and there.

[0007]
If the processing chamber of the recipe sequence of a wafer becomes available, a wafer will be processed next from a degree in order of a wafer sequence list. Although this often functions theoretically, depending on the case, a wafer will be in a "deadlock" condition by the process sequence. This is usually generated, when [to which the process recipe of a specific wafer continues after short-time down stream processing] down stream processing of long duration is included all the time. Although a short-time process process is planned and is completed comparatively promptly, when migration to a processing chamber is tried in the case of a long duration process process, a wafer is interrupted by the destination processing chamber in "under an activity" or processing mode for the same wafer before that. If this occurs, both chambers (a supply source and destination) will be as a matter of fact among the time amount of the die length of a process for a long time in "a condition in use, i.e., the condition for processing after it which cannot be used," for the die length of prolonged processing. The deadlock of the wafer in the chamber of a supply source is carried out literally, and it waits for the destination chamber of a wafer to become available.
[0008]
Also in the case of a short process, this generates an orientation process etc. intrinsically again. Although an orientation process can be performed comparatively promptly, subsequently a wafer waits repeatedly for the following processing chamber in a process sequence to become available within an orientation chamber. This blocks using an orientation chamber for other regularly processible wafers, when prior orientation is carried out and the following processing chamber in a process sequence is subsequently being able to use. This prior orientation becomes economization of most time amount at the subsequent process in a process sequence later.
[0009]
Therefore, in order to reduce or avoid the situation which carried out the deadlock, the approach and equipment which reconstruct the schedule of a system resource are required technically.
[0010]

[Translation done.]

MEANS

[Means for Solving the Problem]
This invention offers the multi chamber processing system equipped with the multitasking control containing the process sequencer which can identify the workpiece or wafer which foresaw and carried out the deadlock of the point of a process sequence. Once the wafer by which the deadlock was carried out is identified, it will be removed from a real-time sequence process by returning it to a load lock preferably by transporting the wafer to the position in readiness in a system. The wafer by which the deadlock was carried out is a load lock, and when the resource becomes available, it waits to be carried out [with which a schedule is reconstructed so that it may be sent to the original destination chamber].
[0011]
Therefore, since a system throughput is increased remarkably, this invention offers the approach and equipment for canceling lock out of the resource of a multi chamber processing system (that is, cluster tool) by using a pre sequencer. With one operation gestalt of this invention, the wafer of the identified dead lock status is a wafer of sequence medium which ended one of the processing process of the, and is waiting to be transported to a destination chamber for the following processing process by the supply source chamber. A destination chamber makes a system resource blockade by making the supply source chamber into the busy condition, while waiting to become available, and probably does not delay other wafers beyond one or it, and a wafer moves the wafer of sequence medium which carried out the deadlock to a position in readiness. For example, a wafer is moved to the internal elevator of a load lock. After migration, a supply source chamber becomes usable, in order to continue processing of other wafers in a system. Subsequently, this invention reconstructs the schedule of the wafer currently held in the position in readiness, when a destination chamber becomes available.
[0012]
With the 2nd operation gestalt of this invention, this invention identifies the wafer with which orientation of [in an orientation chamber] was carried out as a wafer by which the deadlock was carried out. That is, since the 1st processing chamber of the process recipe is using [be / it] it (utilization is impossible), the wafer is not processible. As compared with almost all chemical preparation, for short time amount required for an orientation process, an orientation chamber may serve as main sources of release of a deadlock wafer, and all the throughputs of a system may be reduced. I hear that almost all wafers must pass an orientation chamber, and another reason an orientation chamber becomes a problem has them, and if this does not use an orientation chamber more effectively, it becomes the cause which causes delay. This invention solves this problem by discriminating the wafer with which orientation of [in the orientation chamber which that following destination chamber is using] was carried out from a deadlock wafer, and returning the wafer by which orientation was carried out to a load lock or the convenient wafer position in readiness of a certain others. Thereby, a system continues using an orientation chamber more effectively and it becomes possible to increase a throughput.
[0013]
With the 3rd operation gestalt of this invention, when all processing chambers are using it, this invention carries out orientation of the wafer in the load lock which is waiting to be processed first, and then offers preferably a position in readiness and the prior orientation process returned to a location at those beginning in a load lock. therefore, since the sequence is changed in order to raise the effectiveness of a system even if the orientation process of a specific wafer is performed where of the usual process

sequence, in order to save the processing time, it can come out and perform excluding it.
[0014]
With the operation gestalt which this invention illustrated, these approaches can be performed by the process control sequencer which controls the cluster tool containing the load lock chamber, the migration chamber, the buffer chamber, the orientation chamber, and two or more vacuum processing chambers beyond one or it, for example, the programmed general purpose computer. Each processing chamber is adapted so that one or integrated-circuit processing beyond it (recipe) on a wafer may be performed. The wafer transport station arranged at the core of a migration chamber is used in a wafer in order to move between various chambers, so that a multistage story processing sequence may be promoted.
[0015]
[Embodiment of the Invention]
If the following detailed description is combined with an attached drawing and read, these of this invention and the other objects, a mode, and the description will be understood more clearly, and will more often be explained.
[0016]
In order to promote an understanding, the same element common to a drawing is shown using a same reference figure, when possible.
[0017]
Drawing 1 is the top view of the suitable operation gestalt of the multi chamber semi-conductor wafer processing system 10 (cluster tool) which operates according to this invention. A cluster tool is fitted so that vacuum processing of the workpieces, such as a silicon wafer for the integrated circuits of a very-large-scale-integration (VLSI) mold, may be carried out especially. The cluster tool 10 equips the sealed general target which has eight side attachment walls 13 which demarcate the vacuum migration enclosure or a chamber 14 with the main frame or housing 12 of an octagon.
[0018]
The cluster tool 10 contains four processing chambers (PC1-PC4) 16, 18, 20, and 22, the migration chamber 14, the buffer chamber 28, the wafer orientation chamber / degasifying chamber 30, and one pair of load lock chambers 24 and 26. Each processing chamber expresses the phase or phase from which semi-conductor wafer processing differs. In this disclosure, it is considered that the buffer chamber 28, and the wafer orientation chamber / degasifying chamber 30 are the processing chambers of a special class. Therefore, a chamber is only the vocabulary "a processing chamber" or a thing which includes the chamber of all the forms in an accessible cluster tool according to a transport station.
[0019]
In order to realize wafer migration between these chambers, the migration chamber 14 includes the 1st robot type transport station 82, for example, a single blade robot, (SER). Generally, a wafer 15 is put into the migration cassette 25 made from plastics of a load lock chamber arranged in 24 or 26 one, and is carried in a system from a storage area. The robot type transport station 82 transports every one wafer 15 to a wafer orientation chamber / degasifying chamber 30 from a cassette 25 at once. Generally, the buffer chamber 28 is not used until a wafer is processed within one or the processing chambers 16, 18, and 20 beyond it, and 22. Each wafer is carried and carried by the wafer migration blade 86 arranged in the far edge 84 of the robot device 82. It expands and contracts so that it may be expressed by the arrow head 88, and although the condition of having contracted is shown, this device is rotated so that it may be expressed by the arrow head 90. The orientation of migration is controlled by the control unit 70.
[0020]
A control unit 70 controls processing and wafer migration which are performed by the cluster tool 10. A control device includes exchange circuits, such as memory for storing a microprocessor (CPU) and a control routine and a power unit, a clock circuit, and cache **. A control device 70 also contains I/O peripheral devices, such as a keyboard, a mouse, and a display, again. A control device 70 is a general purpose computer which performs sequencing and scheduling which promote processing and migration of a wafer. The software routine which controls a cluster tool is stored in memory, and in order to promote control of a cluster tool, it is performed by the microprocessor.
[0021]
A control device 70 directs a transport station according to a process schedule, and arranges a wafer to the processing chambers 16, 18, 20, and 22. In order to promote migration of such a wafer, the migration

chamber 14 is enclosed by four processing chambers 16, 18, 20, and 22, and can access them. If processing is completed within a processing chamber, a transport station 82 will move the completed wafer from a processing chamber, and will transport a wafer to the buffer chamber 28. A wafer is taken out from a buffer chamber next and arranged at the load lock chambers 24 or 26.
[0022]
Drawing 2 shows the block diagram of the control device 70 which performs the system process control software for cluster tool 10, and generates automatic control and process sequencing. A control device 70 can be realized in more detail as a general purpose computer (for example, a mainframe computer, a workstation, a personal computer, or a microcomputer) for controlling a multi chamber processing system. A general purpose computer can be equipped with a central processing unit (CPU) or PUROSSESSA 72, memory 71, ROM73 and various I/O devices 74, for example, a monitor, a keyboard, and/or various storage.

[0023]
With a suitable operation gestalt, a control unit is a microcomputer, and it incorporates new system-software application so that it may mention later. The system software is expressed by the software application or module beyond one or it loaded to memory 71 from I/O device 74, for example, the MAG, or an optical disk drive, a diskette, or a tape. The system software is realizable in alternative as firmware stored in read-only memory (ROM) 73 and a prototype. As such a thing, the cluster tool control software of this invention is storable in the medium in which one or the computer read beyond it is possible. Finally, once it loads a software application, a processor 72 will perform this new software in memory, and will realize a cluster tool sequencer. Generally, a processor 72 can be used as the 680x0 mold manufactured, one, for example, Motorola, of the microprocessor of marketing of arbitration.
[0024]
It plans realizable as a circuit device in which some of process processes discussed as a software process here collaborate with a microprocessor in hardware, and various process processes are performed. the process realized by software although the control device is drawn as a general purpose computer programmed to perform various scheduling routines -- application -- it is realizable as hardware, such as a specific integrated circuit (ASIC) or a discrete circuit component. As such a thing, the process process indicated here has the intention of what is interpreted broadly as what is equally performed with the combination of software, hardware, or those arbitration.
[0025]
Sequencer software is multitasking and is divided into the real-time task orientation module called with interruption system from the timer which generates a periodical alerting signal. As shown in drawing 2 , basic sequencer software contains the sequencer task module 200, the screen task module 202, the low-speed task module 204, the chamber task module 206, and the buffer chamber task module 208. When called, each modules 200-208 complete interruption and those processings to processing of a low-speed module more, and return at the exit point of the last of the routine subsequently interrupted. By this conventional multitasking hierarchy actuation, more important actuation can be processed in the real time, without having serious effect for processing on other tasks in a system. When other modules publish the command with which it performs during processing of the module ordered later changing the DS to be used or during the processing to another module by leaving other messages or data for modules respectively to the outlet, as for the above-mentioned modules 200-208, which of other modules can communicate.
[0026]
Generally, the sequencer task module 200 controls sequencing of wafer manufacture or the process for every process including a process recipe program and a washing recipe program. A module 200 operates based on the DS which indicated the sequence and chemical preparation of a process of the wafers loaded into load locks 24 and 26, those processes, and a washing recipe, i.e., various chambers, i.e., a wafer sequence list, (WOL). A process and a washing recipe are respectively related with the specific wafer of WOL through another DS (WMDS), i.e., wafer management data structure, by the pointer to information. The sequencer task module 200 which operates based on the information on WOL and WMS publishes a command to a chamber task module so that the process process of the specific process recipe for a specific chamber or a washing recipe may be performed to right time amount. A wafer moves various chambers with the sequencer task module 200 which operates based on the DS which indicates that migration actuation of a system agrees in the chemical preparation of a specific chamber, i.e., a wafer

migration queue, (WMQ). When a wafer is in WMQ, a sequencer module is ordered to the buffer task module 208 to perform migration by control of the entrance slit bulb of a chamber, and the wafer transport station 82.

[0027]

The chamber task module 206 deals with various recipes and the washing recipes for the integrated-circuit process on a wafer including starting of actual control of a vacuum valve, various ionization processes of a chemical, and RF power source etc. The chamber task module 206 supervises and controls the process which happens by the specialized processing chamber by the actuation based on DS (CDS), i.e., chamber DS.

[0028]

The screen task module 202 makes it possible to leave a message or a command during those activation, before it programs the data of specification [an operator] in the DS of other modules and they are performed by the task module (or other program objects) by the interactive approach. One of the bodies of the screen task module 202 is an interactive text editor, and he is enabled to input data and a command, or for an operator to edit through an operator interface or a display, and to store them in the control for a process, and an administrative module (not shown). The screen task module 202 makes it possible to display a condition with various operators, an alarm, and surveillance intelligence during activation of the automatic-control sequence of a system 10 again.

[0029]

The low-speed task module 204 is also formed in order to supervise the specific task of the system performed comparatively at a low speed and to take timing. For example, the RF coil of a specific chamber was energized or the washing time amount of a chamber can be supervised by determining the amount of the active time amount.

[0030]

Here, the DS used by process control is explained more to a detail in relation to 6 from drawing 3 . WOL DS is indicated by the detail by drawing 3 . Wafer sequence lists are two or more information blocks 1 and 2... It has n and one of the wafers by which the current plan is carried out so that it may be processed into each by the wafer processing system 10 is indicated. Each information block 1, for example, the block displayed as 210, contains the lot number to which the wafer identification number and it which identify a wafer uniquely belong. the wafer management blocks 1 and 2 corresponding to [an information block 210 controls processing of a wafer further, and] the information block 210 of a wafer sequence list, and ... the pointer to WMD (drawing 6) which has n is included further. There are the two flag fields behind these fields. A thing for one to identify a wafer as a process wafer by which the deadlock was carried out, and another are for a deadlock being carried out and identifying it as a wafer (prior orientation) by which orientation was carried out. The field of the last of an information block 210 is a group chamber mask which identifies the group of a processing chamber who can construct the processing schedule of a wafer. The remaining information blocks 2 and 3 of WOL ... n is constituted like an information block 210 and includes the information about the remaining wafer processed with a cluster tool.

[0031]

Wafer management data structure is indicated in detail by drawing 6 , and is structurally similar to the wafer sequence list. WMD -- the wafer management blocks 1 and 2 and ... it has n, and each corresponds to each WOL information block, therefore it corresponds to a specific wafer. The field of the wafer management block 1, for example, the block displayed as 216, contains the lot number and wafer identification number of a wafer. In addition to it, a Status field indicates a process condition. Furthermore, there is program sequence discernment which indicates the washing recipe matched with the process recipe of a wafer and it. The field of the last of the wafer management block 216 records the cassette number and slot of the supply source with which a wafer starts the process. Ideally, such a supply source slot is a slot to which a wafer is returned after processing. The remaining wafer management blocks 2 and 3 of WMDS ... n is constituted like the wafer management block 216, and includes the information about the remaining wafer in a system 10.

[0032]

WMQ DS is indicated in detail by drawing 4 . the identifier blocks 1 and 2 of a large number including the field which identifies the wafer which a wafer migration queue is a list of wafers which need migration, and should move, and its moving trucking (a supply source and destination), and ... it consists of n. The

identifier block 1 displayed as 212 has one and the supply source for memorizing the pointer to the wafer information block in the wafer sequence list which identifies the wafer which should move and a destination processing chamber, or the field including two another fields for identifying the slot of an elevator or a cassette, when a supply source or a destination is a cassette or a load lock. The remaining identifier blocks 2 and 3 of WMQ DS ... n is constituted like the identifier block 212 and contains the same data of the wafer of an and also [migration is the need].
[0033]
Chamber DS is indicated in detail by drawing 5 . Chamber DS is a list of processing chambers and those conditions. chamber DS -- much condition blocks 1 and 2 and ... consisting of n, the condition block 1 displayed as 214 includes the field which identifies the remaining time amount of the recipe process under activation by the chamber in the figure of a meaning, and its chamber. Another field is used as a flag in which it is shown [under an activity of a chamber or] whether it is vacant, and when set, it shows that a chamber is using it and the process recipe process or the washing recipe process is performed. Since the specific chamber has processed the wafer of the number only of which since washing of the last or the number of wafers is stored, the last field is used. The remaining condition blocks 2 and 3 of processing chamber DS ... n is constituted like the condition block 214 and contains the same data about other chambers in a system.
[0034]
Here, in relation to drawing 7 , the sequencer task module 200 containing the pre sequencer constituted according to this invention is explained in more detail. The main program 218 of a sequencer task module is a list of routines performed every 50ms. Generally, the task module 200 is block 220, processes the incoming message from other routines, data, or a command, and starts it by updating the information from other routines which is needed in order to perform the following task.
[0035]
With block 222, **** of a process, and in order to mainly reduce or remove the deadlock of a wafer, a module 200 foresees the point, constructs the migration schedule of the wafer between 30 from the above-mentioned processing chamber 16, and performs the dynamic read-ahead routine or pre sequencer which determines priority. It becomes possible for a system 10 to use a chamber more effectively and to raise the throughput of a cluster tool by this prior sequencing.
[0036]
The dynamic read-ahead program 222 investigates the wafer sequence list used in order that the sequencer task module 200 may set the sequence of processing in order, and determines which wafer within a sequence will cause the problem on a throughput in the future. A program changes the sequence of the wafer discernment data in a wafer sequence list, and a wafer migration queue, and eases or removes these problems from the list of problem wafers with which it was identified within the sequence. As for migration of the wafer for reducing these problems, the correction to be migration before migration of which type of wafer has priority determined by whether a throughput is improved most and performs processing after it is made. Since the pre sequencer routine 222 is performed every 50ms, once a problem is identified, it will be corrected in an instant.
[0037]
After it publishes a command to the buffer chamber task module expressed by block 224 and a wafer moves to those suitable chambers, a command is published by the chamber task module of block 226 so that either the process recipe of a wafer or the washing recipe of a specific chamber may be set in order, so that the sequencer task module 200 after WMQ and WOL, or the required information on other were updated may move a wafer by the sequence beforehand defined as demarcated by WMQ. As such a thing, a chamber task starts with block 228. Once a chamber task starts, the task sequencer module 200 will be dormant with block 229, and will wait for the following alarm clock call.
[0038]
The more detailed functional flowchart of the dynamic read-ahead routine 222 is indicated from drawing 8 a to 8g. It is the general function of the pre sequencer program expressed with block 222 being explained by drawing 8 a in full detail, and the 1st function of the block 230 there foreseeing the point of a wafer sequence list, and looking for a sequence medium deadlock wafer. Although these are ready for completing a part of those processings and moving to the following processing chamber, they are the wafers with which it became clear that the following processing chamber was using it. However, even if they stagnate a

system by not opening the processing chamber or resource with which current arrangement of them is carried out again, they are. This produces the possibility of a deadlock stopped until this and the substantial part of the possible floor to floor time of other wafers become as [receive / the destination chamber of the wafer / a deadlock wafer] (available). The solution of this problem is returning it to the position in readiness of a system 10 preferably at a load lock chamber until it takes out the wafer which carried out the deadlock from that supply source processing chamber and the destination chamber of that wafer becomes available. Thereby, while the deadlock wafer is waiting for the destination chamber, it becomes possible to process other wafers by the opened supply source chamber. A load lock is the example of a position in readiness. A deadlock wafer can be temporarily arranged to the process chamber which is not used or the special position-in-readiness chamber which can be used. A position-in-readiness chamber can include the wafer elevator for storing two or more deadlock wafers which are waiting for the processing chamber.
[0039]
The next function of block 232 is investigating whether there being any wafer [finishing / orientation] which foresaw the point of a wafer sequence list and carried out the deadlock into the orientation chamber 30. Although these are ready for completing orientation and moving to the following processing chamber, they are the wafers with which it became clear that the following processing chamber was using it. However, by not opening for other wafers' processing of the orientation chamber which contains the orientation finishing wafer which carried out the deadlock again, even if they stagnate a cluster tool, they are. This produces the possibility of a deadlock which many of possible processings of the wafer of this and others stop until it comes to be able to perform preparation whose destination chamber receives an orientation finishing wafer. The solution of this problem is the position in readiness of a system 10, and returning to a load lock preferably about it until they take out again the wafer which carried out the deadlock from the orientation chamber 30 and can use a destination chamber. While the orientation finishing wafer which carried out the deadlock is waiting for migration to a destination chamber by this, it becomes possible to carry out orientation of other wafers by the opened orientation chamber. The following two functions in blocks 234 and 236 are the same. They foresee the point of WOL, investigate whether there are any process wafer and returned orientation finishing wafer returned to the position in readiness or the load lock, and reconstruct those schedules for future processing.
[0040]
Once it is attained discernment of a problem wafer (what is easy to become deadlock voice), and that a schedule reconstructs (a) Return the wafer which carried out the deadlock to a load lock, and other wafers are moved to the processing chamber or orientation chamber by which (b) release was carried out. And motion of the wafer transport station 82 for moving the wafer with which the schedule of others containing the load lock wafer which reconstructed the (c) schedule was carried out can attach priority by examining the class of wafer with block 238. The class of wafer contains the wafer which carried out the deadlock, the wafer which carried out orientation, the wafer of sequence medium, and a prototype. A trial shows clearly what kind of wafer is listed by the wafer migration queue, and, subsequently is reorganized based on the class of wafer which wafer discernment data should move. For example, although the wafer of sequence medium which carried out the deadlock is orientation ending, it has priority higher than a raw wafer. Furthermore, although it is orientation ending, a raw wafer has priority higher than the wafer which must still be picked out from a load lock, although orientation is not carried out. After attaching priority to a wafer migration queue, the wafer within a queue moves to each of those destination as it is shown in block 240.
[0041]
From now on, the detail routine which realizes the block 230 of drawing 8 a in order to identify the wafer of sequence medium in which the deadlock was carried out by the identification read ahead is explained in more detail in relation to drawing 8 b. Although all the wafers in a wafer sequence list are inspected by the routine of drawing 8 b before they return to main routine, they indicate the process of only one wafer for **** to a term ** sake. It is determined whether a routine has the destination chamber of a large number which can read chamber discernment of the chamber (supply source chamber) containing the present wafer first with block 242, and can receive a wafer at the process of whether it is a group chamber and a degree that is,. If a chamber is not a group chamber, a routine will use the found specific chamber ID number, and will progress to block 246. When chamber discernment shows that the wafer is contained in a

group chamber, a chamber available next is chosen in a group by reading the remaining time amount of a recipe with block 244 by investigating the chamber DS of each chamber of the group. A program continues discernment of a supply source chamber, and discernment of a destination chamber within block 246, and it is determined whether a destination chamber is available to the next migration. The availability of a destination chamber is examined by the busy status flag of chamber DS, and the residual time of the recipe field.
[0042]
When a chamber is available, wafer discernment data are inputted into wafer migration queue DS with block 252, and if it is the entry of the last of the wafer sequence list which this should investigate, a program will return. When a chamber cannot be used, with block 250, a deadlock flag is set to the specific wafer, and discernment of the following processing chamber which is under activity now is saved on a wafer sequence list. Subsequently to a load lock the destination chamber field of the wafer is changed, and wafer discernment data are inputted into a wafer migration queue with block 252. As such a thing, a wafer moves to a load lock after activation of a buffer chamber task module. If all the wafers of WOL are examined, a routine will return to a main task module after that.
[0043]
The detailed routine which realizes the block 232 of drawing 8 a in order to identify the orientation finishing wafer in which the deadlock was carried out by the dynamic read ahead is explained in more detail in relation to drawing 8 c. Although all the wafers of a wafer sequence list are inspected by the routine of drawing 8 c before they return to main routine, they indicate the process of only one wafer for **** to a term ** sake. It is determined whether a routine has the destination chamber of a large number which can read chamber discernment of the chamber (supply source chamber) containing the present wafer first with block 254, and can receive a wafer at the process of whether it is a group chamber and a degree that is,. If a chamber is not a group chamber, a routine will use the found specific chamber ID number, and will progress to block 258. When chamber discernment shows that the wafer is contained in a group chamber, a chamber available next is chosen in a group by reading the remaining time amount of a recipe with block 256 by investigating the chamber DS of each chamber of the group. A program continues discernment of a supply source chamber, and discernment of a destination chamber within block 258, and a destination chamber determines whether it is available as the migration to which the schedule of the degree was carried out. The availability of a destination chamber is examined by the busy status flag and the recipe residual time field of chamber DS.
[0044]
It examines with block 260, when the following chamber is available, wafer discernment data are inputted into wafer migration queue DS with block 264, and if it is the entry of the last of the wafer sequence list which this should investigate, a program will return. When a chamber cannot be used, with block 262, a prior orientation flag is set to the specific wafer, and discernment of the following processing chamber which is under activity now is saved on a wafer sequence list. Subsequently to a load lock the destination chamber field of the wafer is changed, and wafer discernment data are inputted into a wafer migration queue with block 264. As such a thing, a wafer moves to a load lock after activation of a buffer chamber task module. If all the wafers of WOL are examined, a routine will return to the block 232 of drawing 8 a after that.
[0045]
After processing a sequence medium wafer and an orientation finishing wafer, it moves from a routine to drawing 8 d, and a routine does not have the returned process wafer and the returned orientation finishing wafer, or inspects a wafer sequence list, and reconstructs those schedules towards the original destination chamber here. The routine of drawing 8 d follows and realizes the process of the approach expressed by the blocks 234 and 236 of drawing 8 a. About all the wafers in a wafer sequence list, a routine performs the group of steps 266-278, and starts him by examining whether the information block by which is block 266 and current addressing is carried out by the wafer sequence list identifies the 1st wafer. Next, a routine opts for discernment of the chamber in which the wafer by which current addressing is carried out is located with block 268. Chamber ID is examined with block 270 next, and it is determined whether it is a load lock chamber. When that is not right, a program moves to the following wafer by moving to the trial of block 278. However, it may be the returned wafer when a wafer is in a load lock. A sequence medium flag is examined with block 272, an orientation finishing flag is examined with block 274, and it is determined whether either is set or not. When one of flags is set, this is a returned wafer which is waiting for a

schedule to reconstruct. Moving from the following process to drawing 8 e, this realizes the schedule **** repair routine called by block 276. When it is shown by the trial of blocks 272 and 274 that the deadlock flag is not set, as for a program, the next information block of a wafer sequence list is inspected by blocks 278 and 280 return and there. After all the wafers of a wafer sequence list are inspected, a program returns to the block 234 of drawing 8 a, or 236 then.

[0046]

Drawing 8 e shows the flow chart of the routine which realizes the wafer schedule **** repair process of the block 276 of drawing 8 d. It is determined whether a routine has the destination chamber of a large number which can read chamber discernment of the chamber (supply source chamber) containing the present wafer first with block 282, and can receive a wafer at the step of whether it is a group chamber and a degree that is,. The identification information of a group chamber is contained in a group mask. If a chamber is not a group chamber, a routine will use the found specific chamber ID number, and will progress to block 286. When chamber discernment shows that the wafer is contained in a group chamber, a chamber available next is chosen in a group by reading the residual time of a recipe with block 284 by investigating the chamber group mask and the chamber DS of each chamber (that is, when the close chamber is in a group mask). With block 286, a routine inspects chamber DS and it is determined whether a destination chamber (target) is available. It refers for whether the target chamber of a routine is available with block 288. When enquiry is answered in the negative, a routine returns to the block 276 of drawing 8 d. Therefore, a wafer does not rehave a schedule constructed but it must wait for it so long. When enquiry of block 288 is answered in suitable, a routine goes into block 290 and updates WMQ DS. As such a thing, a wafer moves to the chamber for processing from the position in readiness of a load lock after activation of the following buffer chamber task. Next, a routine returns to the block 276 of drawing 8 e.

[0047]

After identifying the sequence intermediate-processing-intermediate-treatment wafer by which the deadlock was carried out, and the orientation finishing wafer by which the deadlock was carried out, reconstructing the schedule of the migration, and identifying the returned sequence medium wafer and the wafer by which prior orientation was carried out and reconstructing the schedule of the migration, a program determines the priority of a wafer migration queue. The functional flowchart where the step of this process is detailed is described in more detail to drawing 8 f which realizes the block 238 of drawing 8 a. A wafer migration queue begins a process and a washing recipe, a pre sequencer, a screen task module (operator control), and a prototype, and is loaded in random sequence from various programs with the need of moving a wafer. The routine of drawing 8 f redoes sequencing of these migration, attaches priority to **** which may be generated in a process, and prevents the lock out. Sequencing of a wafer sequence list is redone and the raw wafer in a load lock is put on a wafer [finishing / the sequence medium when the sequence medium and orientation finishing wafer by which was set to the second in the washing wafer, and the deadlock was carried out in the first place / of priority / to the third in the processed wafer was returned in the load lock by the fourth, and orientation], and the fifth. Since the processing chamber which needs washing is not stopped by this approach, that chamber becomes available by the time amount in which the shortest is possible. Furthermore, the completed process wafer can be picked out from a system, or can be moved to the following step, and a resource can be released by it for other wafers which should move by the second priority. A wafer [finishing / the sequence medium by which the deadlock was carried out, or orientation] moves by the third priority, and releases a specific processing chamber or a specific orientation chamber, respectively. Subsequently, the returned orientation finishing wafer and the returned sequence medium wafer move using the interval of a schedule which may happen, and completes processing in these wafers. Finally, the raw wafer in a load lock is the tail end of a migration queue. Thereby, a new wafer does not create a new schedule until other wafers processed selectively are processed. The minimum migration of a wafer is planned by this approach, and through processing effectiveness improves remarkably.

[0048]

As a matter of fact, the program of drawing 8 g generates a command to a buffer chamber task module so that a wafer may be moved, and it realizes the block 240 of drawing 8 a as such a thing. After redoing sequencing of WMQ, migration of the wafer of a list is started. Sequential addressing of each entry of a wafer migration queue is carried out with block 302, and a command is taken out by the buffer task module so that the specific wafer for [current] an interest may be moved. a ******** [that a routine is block

304 and it is a wafer / finishing / the sequence medium to which the wafer was returned by investigating wafer discernment data with the flag of a wafer sequence list, or orientation / after the wafer migration command to the wafer of a list is taken out] -- and it is determined whether the supply source chamber of the wafer is a load lock. When both conditions are fulfilled, it was determined as what kind of wafer, or (that is, is a wafer the wafer by which the deadlock was carried out, and a wafer by which prior orientation was carried out also to it?) a prior orientation flag or a deadlock flag is reset by the wafer sequence list with block 306. After clearing these flags, a program is block 308 and it is determined whether this is the wafer of the last of a wafer migration queue. If it is not the last wafer, a program will publish return and another wafer migration command to block 302. After the last wafer command is published, a program returns to the block 240 of drawing 8 a.

[0049]

The buffer chamber task module 208 explained further in full detail in the functional flowchart of drawing 9 a to 9b is a program which controls the wafer transport station 80 and performs migration of the wafer between a processing chamber and a load lock. The buffer chamber task module 208 is called with interruption system every 20ms, and performs this actuation together with the task of others relevant to a buffer chamber. When it is determined that the buffer chamber task module 208 will have the command which moves a specific wafer with block 310, it calls the wafer migration program 312, in order to attain the task. As the above-mentioned [the wafer migration program 312], a transport station 82 lifts a wafer from the supply source chamber, and controls moving it to the destination chamber. This program controls closing motion of the entrance slit valve of a chamber again, makes migration possible if needed, and attains the separation under processing and washing. After a program receives a wafer migration command according to another description of a pre sequencer, it examines whether a destination is a slot in a cassette. When this trial is truth, a pre sequencer foresees the point of a wafer migration queue, determines whether there is any another wafer connected to the load lock, and makes the slit valve of a load lock as [open beam] after exchange of a wafer with block 316. Two or more wafer exchange can be attained as such a thing, without opening and closing the slit valve of a load lock at every exchange. Then, after a routine progresses to block 317 and completes other tasks there, it returns at the interruption exit point.

[0050]

The detail chart of the routine of drawing 9 b realizes step 316 of drawing 9 a. First, it is block 320, a program determines whether a wafer is in a wafer migration queue more, and if there is nothing, it will return a negative indicator to the wafer migration program 312. If it is, the following wafer will be examined with blocks 322 and 324, and it will be investigated whether the supply source cassette discernment is equal to the wafer cassette discernment for which it comes on the contrary. When a trial is answered in suitable, a routine determines the current location of the following wafer there following block 326. After determining the current location of a wafer, it examines whether a program is equal to the cassette discernment for which it is [discernment] block 328 and the location comes on the contrary. In the case of yes, a program returns an affirmative response to a wafer migration program. When that is not right, it investigates whether a program relapses into block 320, examines a wafer migration queue there, and has a wafer more. The flag in which it is shown that the wafer by which the schedule was carried out is in a wafer migration list is generated by the wafer migration program so that it may be arranged at the load lock and may take out from a load lock by this, and the schedule of the current wafer is carried out so that it may move to a load lock. A slit valve door must be opened in order that close may move the wafer to which it comes to a load lock, and it is intentionally made as [open beam] so that there may be no need of closing only in order to transport after that the wear which a wafer migration program leaves and to open again later. Since there is the most amount of wafer migration in a load lock, with prior sequencing, especially this refined time amount economization raises the effectiveness of a load lock mechanism, and falls wear.

[0051]

Although this invention was explained in relation to the suitable operation gestalt, it does not mean restricting this description to the specific gestalt which specified the range of this invention, and has the intention of covering all the alternative examples included in the pneuma of this invention demarcated by the claim reversely [that], and within the limits, a modification, and an equal object.

[Brief Description of the Drawings]

[Drawing 1]

It is the outline top view of the multi chamber processing system which operates according to this invention.
[Drawing 2]
It is the pictorial system flow chart of the process control program of the control unit shown in drawing 1 .
[Drawing 3]
It is the tabular format expression of the wafer sequence list data structure used by the process control program shown in drawing 2 .
[Drawing 4]
It is the tabular format expression of the wafer migration queue DS used by the process control program shown in drawing 2 .
[Drawing 5]
It is the tabular format expression of the chamber DS used by the process control program shown in drawing 2 .
[Drawing 6]
It is the tabular format expression of the wafer management data structure used by the process control program shown in drawing 2 .
[Drawing 7]
It is the detail chart of the sequencer task module of the process control program shown in drawing 2 .
[Drawing 8 a]
It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .
[Drawing 8 b]
It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .
[Drawing 8 c]
It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .
[Drawing 8 d]
It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .
[Drawing 8 e]
It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .
[Drawing 8 f]
It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .
[Drawing 8 g]
It is the detail chart of the read-ahead capability of the sequencer task module shown in drawing 7 .
[Drawing 9 a]
It is the detail chart of the buffer chamber task module of the process control program shown in drawing 2 .
[Drawing 9 b]
It is the detail chart of the buffer chamber task module of the process control program shown in drawing 2 .

[Translation done.]

(19)日本国特許庁（ＪＰ）　(12) 公表特許公報（Ａ）　(11)特許出願公表番号
特表2002－506285
（P2002－506285A）
(43)公表日　平成14年2月26日(2002.2.26)

(51)Int.Cl.⁷	識別記号	ＦＩ	テーマコード゛(参考)
Ｈ０１Ｌ　21/68		Ｈ０１Ｌ　21/68	Ａ　５Ｆ０３１

審査請求　未請求　予備審査請求　有　（全 44 頁）

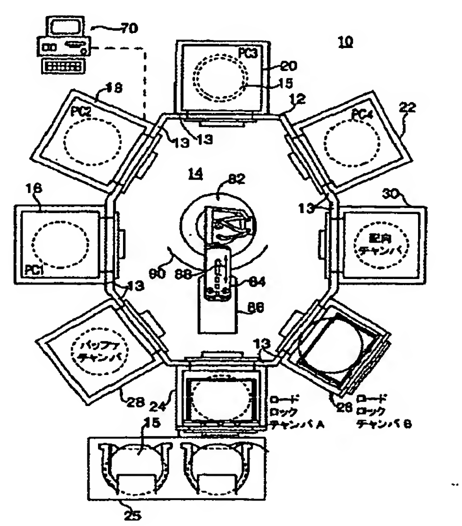
(21)出願番号　特願2000－535034(P2000－535034)
(86)(22)出願日　平成11年2月11日(1999.2.11)
(85)翻訳文提出日　平成12年9月4日(2000.9.4)
(86)国際出願番号　ＰＣＴ／ＵＳ９９／０３１１３
(87)国際公開番号　ＷＯ９９／４５５７５
(87)国際公開日　平成11年9月10日(1999.9.10)
(31)優先権主張番号　０９／０３３，８２２
(32)優先日　平成10年3月3日(1998.3.3)
(33)優先権主張国　米国（ＵＳ）
(81)指定国　ＪＰ

(71)出願人　アプライド　マテリアルズ　インコーポレイテッド
ＡＰＰＬＩＥＤ　ＭＡＴＥＲＩＡＬＳ，ＩＮＣＯＲＰＯＲＡＴＥＤ
アメリカ合衆国　カリフォルニア州　95054　サンタ　クララ　バウアーズ　アベニュー　3050
(72)発明者　ヌグイェン，　チュー
アメリカ合衆国，　カリフォルニア州，　サン　ノゼ，　シルヴァー　リーフ　ロード　5911
(74)代理人　弁理士　長谷川　芳樹　（外1名）

最終頁に続く

(54)【発明の名称】　マルチチャンバ半導体ウェハ加工システムでウェハを順序付けるための方法および装置

(57)【要約】
マルチチャンバ半導体ウェハ加工システムを制御するための方法および装置である。この加工システム10は、移送チャンバ14の周囲に複数の処理チャンバ16, 18, 20, 22を含む。中央に配置されたウェハ移送機構82は、処理チャンバ間のウェハの移動を達成する。プロセスシーケンサコントロールは、加工における遅延を防止するためにプレシーケンサまたは先読み機能を有する、実時間マルチタスク制御プログラムである。１つの実施形態では、先読み機能は、行先チャンバが使用中であるためにそれ以上加工することができないシーケンス中間または配向済みウェハを識別する。行先チャンバが利用可能になるまで待ってシステム資源を浪費するのではなく、ウェハを待機位置、好ましくはロードロックに移送して、最も早い時間にその加工を完了するようにスケジュールを組み直す。



【特許請求の範囲】

【請求項１】　半導体ウェハ加工システムであって、

複数の処理チャンバと、

前記複数の処理チャンバの間の中心に配置され、処理チャンバ間でウェハを移送する移送機構を含む移送チャンバと、

処理チャンバ間のウェハの移送のシーケンスのタイミングを合わせて順序付けるように前記ウェハ転送機構を制御し、かつデッドロックされたウェハのスケジュールを組み直すためのシーケンスタスク制御装置と

を備える半導体ウェハ加工システム。

【請求項２】　前記シーケンスタスク制御装置が、

供給源処理チャンバ内でデッドロックされたウェハを識別するための手段と、

前記識別されたウェハを待機位置に移送し、それによって前記供給源処理チャンバを解放するように、ウェハ移送機構に命令するための手段と

をさらに備える、請求項１に記載のシステム。

【請求項３】　デッドロックされたウェハを識別するための前記手段が、

ウェハの行先チャンバを決定し、その供給源処理チャンバから意図された行先処理チャンバへの移送のスケジュール時にその資源が利用可能かどうかを決定するための手段と、

行先処理チャンバが利用不能なときには必ずそのウェハをデッドロックされたウェハと識別するための手段と

をさらに備える、請求項２に記載のシステム。

【請求項４】　デッドロックされたウェハを識別するための前記手段が、

その資源が利用可能になったときに、前記デッドロックされたウェハを待機位置からその意図された行先チャンバに移送するスケジュールを組むように、ウェハ加工シーケンスを順序変更するための手段をさらに備える、請求項３に記載のシステム。

【請求項５】　前記待機位置がロードロック内にある、請求項４に記載のシステム。

【請求項６】　前記デッドロックされたウェハがシーケンス中間のウェハで

ある、請求項１に記載のシステム。

【請求項７】　配向チャンバをさらに含み、

前記デッドロックされたウェハが配向チャンバである、請求項１に記載のシステム。

【請求項８】　前記シーケンスタスク制御装置がウェハ移動待ち行列、チャンバデータ構造、ウェハ管理データ構造、およびウェハ順序リストを使用する、請求項１に記載のシステム。

【請求項９】　前記ウェハ順序リストが、ウェハ識別データを用いて前記複数のウェハ内の各ウェハを一意に識別する情報を含む、請求項８に記載のシステム。

【請求項１０】　前記ウェハ移動待ち行列が、前記複数のウェハ内の前記ウェハの各々の供給源チャンバおよび行先チャンバを識別する情報を含む、請求項８に記載のシステム。

【請求項１１】　前記チャンバデータ構造がチャンバ加工パラメータに関する情報を含む、請求項８に記載のシステム。

【請求項１２】　前記ウェハ管理データ構造が、各ウェハの供給源、各ウェハの状態、供給源カセット識別子、および供給源スロット識別子を識別する情報を含む、請求項８に記載のシステム。

【請求項１３】　マルチチャンバウェハ加工システムでウェハを加工する方法であって、

デッドロックされたウェハを識別する工程と、

前記デッドロックされたウェハの行先チャンバが利用可能になるまで、前記デッドロックされたウェハを待機位置に移動する工程と、

前記デッドロックされたウェハを前記行先チャンバに移動する工程と

を含む、マルチチャンバウェハ加工システムでウェハを加工するための方法。

【請求項１４】　前記デッドロックされたウェハが配向チャンバ内にある、請求項１３に記載の方法。

【請求項１５】　前記識別工程が、

ウェハ移動待ち行列を調べて、特定のウェハの行先チャンバを識別する工程と

、

チャンバデータ構造を調べて、前記行先チャンバが利用可能かどうかを決定する工程と、

行先チャンバが利用できない場合には、特定のウェハをデッドロックと識別する工程と

をさらに含む、請求項１３に記載の方法。

【請求項１６】　前記行先チャンバが利用可能になったときを決定する工程と、

特定のウェハを行先チャンバに移動するためのスケジュールの組直しを促進するために、特定のウェハ識別データでウェハ移動待ち行列を更新する工程と

をさらに含む、請求項１５に記載の方法。

【請求項１７】　前記ウェハ移動待ち行列が更新された後、前記ウェハ移動待ち行列内のいずれかのウェハに優先順位を付ける工程をさらに含む、請求項１６に記載の方法。

【請求項１８】　前記デッドロックされたウェハが配向チャンバを供給源チャンバとし、前記方法がさらに、複数のウェハがロードロックから配向チャンバに順次移送され、行先チャンバが利用可能になるまでロードロックに戻される間、配向チャンバの扉を開いたままに維持する工程を含む、請求項１３に記載の方法。

【請求項１９】　配向されたウェハを前記ロードロックから前記行先チャンバへ直接移動させる工程をさらに含む、請求項１８に記載の方法。

【請求項２０】　各ウェハがプロセスレシピに従って１つまたはそれ以上の集積回路プロセスに適応した複数の処理チャンバ内で順次加工され、複数のウェハのスケジュールが組まれる、ウェハをマルチチャンバ加工する方法であって、

前記複数のウェハが前記複数の処理チャンバおよび異なるプロセスで加工されるタイミングおよび順序を識別するウェハ順序リストを組み立てる工程と、

前記ウェハ順序リストに従って前記ウェハを加工する工程と、

ウェハ順序リストで識別される各ウェハについて、供給源処理チャンバにおけるその現在の状態およびそのプロセスレシピから、その処理シーケンスの次の行

先処理チャンバおよび現在のプロセス工程の完了時の行先処理チャンバの状態を決定する工程と、

その状態が利用可能である場合には、ウェハを行先処理チャンバに移送する工程と、

行先処理チャンバが利用不能状態である場合には、ウェハを待機チャンバに移送する工程と、

待機チャンバに移送されたウェハについて、ウェハ処理シーケンスのスケジュールを組み直すためにウェハ順序リストを変更する工程と

を含む方法。

【請求項２１】　ウェハ順序リスト内のウェハのいずれかが移送できるかどうかを決定する工程と、どのウェハも移送できない場合には、

ロードロックおよび配向チャンバの両方を移送チャンバに対して開く工程と、

ウェハを前記ロードロックから前記配向チャンバに移送し、前記ウェハを前記配向チャンバ内で配向し、前記配向済みウェハを前記ロードロックに移送することによって、ウェハの配向を実行する工程と、

行先チャンバが利用可能になるまで、前記ウェハ配向を前記ロードロック内の他のウェハに対して繰り返す工程と、

前記ロードロックおよび配向チャンバを閉じ、ウェハを行先チャンバに移送する工程と

をさらに含む、請求項２０に記載の方法。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、集積回路ウェハ加工システムに関し、さらに詳しくは、複数の処理チャンバを有するマルチチャンバ半導体ウェハ加工システムに関する。

【０００２】

【従来の技術】

半導体ウェハなどの加工物に複数の工程を順次かつ同時に実行することができるマルチチャンバ半導体ウェハ加工システム（クラスタツールとして知られる）は、Ｍａｙｄａｎらに発行された米国特許第４，９５１，６００号（Ｍａｙｄａｎ　Ｉ）、および同じくＭａｙｄａｎらに発行された米国特許第５，２９２，３９３号（Ｍａｙｄａｎ　ＩＩ）に示されており、これらの開示を参照によって本願明細書に組み入れる。

【０００３】

Ｍａｙｄａｎ　ＩおよびＭａｙｄａｎ　ＩＩに記載されたクラスタツールは基本的に、１つまたはそれ以上のロードロックチャンバと、移送チャンバと、各チャンバの選択的に閉鎖可能なスリットを介してロードロックチャンバおよび移送チャンバに連絡している複数の真空処理チャンバとを含む統合真空加工システムである。各々の処理チャンバは、チャンバ内に配置された１つまたはそれ以上のウェハの化学気相堆積、物理的蒸着、エッチ処理、および急速熱アニールなど、１つまたはそれ以上の集積回路処理を実行するように適応させることができる。ロードロックチャンバは、ウェハをロードロックチャンバの入口に配置するように適応させた外部ウェハエレベータと、ロードロックチャンバの出口に隣接する位置または移送位置にウェハを移動させるように適応させた内部エレベータとを組み込んでいる。

【０００４】

中央に配置されたウェハ移送機構は移送チャンバ内に取り付けられ、一般的に水平なウェハ保持ブレードを含む。ウェハ移送機構はブレードの回転、伸長、および引込みを実行して、ブレードを選択的に外部エレベータ、内部エレベータ、

および処理チャンバに配置する。

【０００５】

通常の加工モードでは、プロセス制御装置がウェハを外部エレベータの移送カセット内のスロットから、１つのロードロックの入口を通して内部エレベータの保持板に移送する。クラスタツールにウェハが積載された後、ロードロックの入口が密閉される。次いで、ロードロック、移送チャンバ、および処理チャンバは加工用真空レベルにまでポンプで排気される。実時間自動プロセスシーケンサが起動して、第１ウェハを処理チャンバに移動し、その入口を閉じてその加工を開始し、第２ウェハを処理チャンバに移動し、その入口を閉じてその加工を開始する、等。それにより、クラスタツールが閉鎖され真空状態にある間の処理チャンバ間のウェハの経路を計画することによって、複数のウェハのマルチチャンバ連続シーケンシャルプロセスが達成される。ウェハの加工が完了すると、それは通常、特定のロット内の全てのウェハの加工が完了して戻されるまで、内部エレベータの特定の板に戻される。次いでクラスタツールは大気に開放され、ウェハが外部エレベータの移送カセットに取り出される。

【０００６】

マルチチャンバ加工システムの要素の化学的処理の調整および制御は、対話型ユーザ入力およびシステム監視を可能にする実時間マルチタスク制御プログラムによって提供される。一般的に、シーケンサタスクモジュールは、内部エレベータ保持板上の各ウェハの識別およびプロセスレシピまたはシーケンスを含むウェハ順序リストを読み取り、処理チャンバ間のウェハの移送およびそこで行われる化学的処理を計画する。

【０００７】

ウェハのレシピシーケンスの処理チャンバが利用可能になると、ウェハは、ウェハ順序リストの順番に次から次に処理される。これは理論的にはよく機能するが、場合によっては、ウェハがプロセスシーケンスで「デッドロック」状態になる。これは通常、特定のウェハのプロセスレシピが短時間処理工程の後に続くずっと長時間の処理工程を含む場合に発生する。短時間プロセス工程は計画され比較的迅速に完了するが、長時間プロセス工程の場合、処理チャンバへの移送が試

みられるときに、ウェハは、その前の同様のウェハのために「使用中」または加工モードの行先処理チャンバによって遮られる。これが発生すると、長時間処理の長さのため、両方のチャンバ（供給源と行先）が長時間プロセスの長さの時間中、事実上「使用中」の状態、つまりそれ以後の加工のために使用不能な状態になる。供給源のチャンバ内のウェハは文字通りデッドロックされ、ウェハの行先チャンバが利用可能になるのを待つ。

【０００８】

これはまた、配向プロセスなど、本質的に短いプロセスの場合にも発生する。配向プロセスは比較的迅速に実行することができるが、次いでウェハは、プロセスシーケンスにおける次の処理チャンバが利用可能になるのを配向チャンバ内で何回も待つ。これは、事前配向し、次いでプロセスシーケンスにおける次の処理チャンバが利用できるになったときに規則的に加工することができる他のウェハのために配向チャンバを使用することを妨害する。この事前配向は後で、プロセスシーケンスにおけるその後の工程でかなりの時間の節約になる。

【０００９】

したがって、デッドロックした状況を低減または回避するためにシステム資源のスケジュールを組み直す方法および装置が技術上必要である。

【００１０】

【課題を解決するための手段】

本発明は、プロセスシーケンスの先を見通してデッドロックした加工品またはウェハを識別することができるプロセスシーケンサを含むマルチタスク制御を備えたマルチチャンバ加工システムを提供する。デッドロックされたウェハは、ひとたび識別されると、そのウェハをシステム内の待機位置に移送することによって、好ましくはそれをロードロックに戻すことによって、実時間シーケンスプロセスから除去する。デッドロックされたウェハは、ロードロックで、その資源が利用可能になったときにその当初の行先チャンバに送られるようにスケジュールが組み直されるされるのを待つ。

【００１１】

したがって、本発明は、システムのスループットを著しく増加するために、プ

レシーケンサを使用することによってマルチチャンバ加工システム（つまりクラスタツール）の資源の閉塞を解除するための方法および装置を提供する。本発明の１つの実施形態では、識別されたデッドロック状態のウェハは、その加工工程の１つを終了したシーケンス中間のウェハであり、その次の加工工程のために行先チャンバに移送されるのを供給源チャンバで待っている。ウェハが行先チャンバが利用可能になるのを待っている間、供給源チャンバを使用状態にしておくことによってシステム資源を閉塞させて、おそらく１つまたはそれ以上の他のウェハを遅延させるのではなく、デッドロックしたシーケンス中間のウェハを待機位置に移動させる。例えば、ウェハをロードロックの内部エレベータに移動させる。移動後、供給源チャンバはシステム内の他のウェハの加工を続けるために使用可能になる。次いで、本発明は、行先チャンバが利用可能になったときに、待機位置に保持されているウェハのスケジュールを組み直す。

【００１２】

本発明の第２実施形態では、本発明は、配向チャンバ内の配向されたウェハをデッドロックされたウェハとして識別する。つまり、そのウェハは、そのプロセスレシピの第１処理チャンバが使用中（利用不能）なので加工することができない。ほとんどの化学的処理に比較して配向プロセスに必要な短い時間のため、配向チャンバはデッドロックウェハの主要な発生源となり、システムの全スループットを低下させる可能性がある。配向チャンバが問題になる別の理由は、ほとんどのウェハが配向チャンバを通過しなければならないということであり、これは、配向チャンバをもっと効果的に利用しなければ、遅延を引き起こす原因になる。本発明は、その次の行先チャンバが使用中である配向チャンバ内の配向されたウェハをデッドロックウェハと識別し、配向されたウェハをロードロックまたは何らかのその他の便利なウェハ待機位置に戻すことによって、この問題を解決する。これにより、システムは配向チャンバをより効果的に使用し続け、スループットを増加することが可能になる。

【００１３】

本発明の第３実施形態では、全ての処理チャンバが使用中である場合、本発明は、加工されるのを待っているロードロック内のウェハを最初に配向し、次に待

機位置、好ましくはロードロック内のそれらの当初に位置に戻す、事前配向プロセスを提供する。したがって、通常のプロセスシーケンスのどこで特定のウェハの配向プロセスが行われたとしても、シーケンスはシステムの効率を高めるために変更されているので、処理時間を節約するためにそれを省くことがでできる。

【００１４】

本発明の図示した実施形態では、これらの方法は、１つまたはそれ以上のロードロックチャンバ、移送チャンバ、バッファチャンバ、配向チャンバ、および複数の真空処理チャンバを含むクラスタツールを制御するプロセス制御シーケンサ、例えばプログラムされた汎用コンピュータによって、実行することができる。各々の処理チャンバは、ウェハ上の１つまたはそれ以上の集積回路加工（レシピ）を実行するように適応される。移送チャンバの中心に配置されたウェハ移送機構は、多段階加工シーケンスを促進するように、ウェハを様々なチャンバの間を移動させるために使用される。

【００１５】

【発明の実施の形態】

本発明のこれらおよびその他の目的、態様、および特徴は、以下の詳細な記述を添付の図面と併せて読むと、より明確に理解され、よりよく説明されるであろう。

【００１６】

理解を促進するために、図面に共通する同一の要素は、可能な場合、同一参照数字を用いて示す。

【００１７】

図１は、本発明に従って作動するマルチチャンバ半導体ウェハ加工システム１０（クラスタツール）の好適な実施形態の平面図である。クラスタツールは特に、超大規模集積（ＶＬＳＩ）型の集積回路用のシリコンウェハなどの加工品を真空加工するように適応させる。クラスタツール１０は、真空移送エンクロージャまたはチャンバ１４を画定する８つの側壁１３を有する、密閉された一般的に八角形のメインフレームまたはハウジング１２を備えている。

【００１８】

クラスタツール１０は、例えば４つの処理チャンバ（ＰＣ１－ＰＣ４）１６、１８、２０、２２、移送チャンバ１４、バッファチャンバ２８、ウェハ配向チャンバ／脱ガスチャンバ３０、および１対のロードロックチャンバ２４および２６を含む。各処理チャンバは、半導体ウェハ加工の異なる段階または相を表す。この開示では、バッファチャンバ２８およびウェハ配向チャンバ／脱ガスチャンバ３０は、特殊な種類の処理チャンバとみなす。したがって、用語「処理チャンバ」または単にチャンバとは、移送機構によってアクセス可能なクラスタツール内の全ての形のチャンバを包含するものである。

【００１９】

これらのチャンバ間のウェハ移送を実現するために、移送チャンバ１４は、第１ロボット式移送機構８２、例えば単一ブレードロボット（ＳＥＲ）を含む。ウェハ１５は一般的に、ロードロックチャンバの１つ２４または２６内に配置されたプラスチック製移送カセット２５に入れて、貯蔵所からシステム内に運ばれる。ロボット式移送機構８２は、ウェハ１５を一度に１つずつ、カセット２５からウェハ配向チャンバ／脱ガスチャンバ３０へ移送する。バッファチャンバ２８は一般的に、ウェハが１つまたはそれ以上の処理チャンバ１６、１８、２０、２２内で処理されるまで使用されない。個々のウェハは、ロボット機構８２の遠端８４に配置されたウェハ移送ブレード８６に載せて運ばれる。この機構は、収縮した状態が示されているが、矢印８８によって表されるように伸縮し、かつ矢印９０によって表されるように回転する。移送の配向は、制御装置７０によって制御される。

【００２０】

制御装置７０は、クラスタツール１０によって実行される加工およびウェハ移送を制御する。制御装置はマイクロプロセッサ（ＣＰＵ）、制御ルーチンを格納するためのメモリ、および電源装置、クロック回路、キャッシュ等などの支援回路を含む。制御装置７０はまた、キーボード、マウス、およびディスプレイなどの入出力周辺装置をも含む。制御装置７０は、ウェハの加工および移送を促進する順序付けおよびスケジューリングを実行する汎用コンピュータである。クラスタツールを制御するソフトウェアルーチンはメモリに格納され、クラスタツール

の制御を促進するためにマイクロプロセッサによって実行される。

【００２１】

制御装置７０は、プロセススケジュールに従って移送機構を指示し、ウェハを処理チャンバ１６、１８、２０、および２２に配置する。そのようなウェハの移動を促進するために、移送チャンバ１４は、４つの処理チャンバ１６、１８、２０、および２２によって取り囲まれ、それらにアクセスすることができる。処理チャンバ内で加工が完了すると、移送機構８２は完了したウェハを処理チャンバから移動させ、ウェハをバッファチャンバ２８に移送する。ウェハは次にバッファチャンバから取り出され、ロードロックチャンバ２４または２６に配置される。

【００２２】

図２は、クラスタツール１０用のシステムプロセス制御ソフトウェアを実行して自動制御およびプロセス順序付けを生成する制御装置７０のブロック図を示す。さらに詳しくは、制御装置７０は、マルチチャンバ加工システムを制御するための汎用コンピュータ（例えばメインフレームコンピュータ、ワークステーション、パーソナルコンピュータ、またはマイクロコンピュータ）として実現することができる。汎用コンピュータは、中央処理装置（ＣＰＵ）またはプロッセッサ７２、メモリ７１、ＲＯＭ７３、および様々な入出力装置７４、例えばモニタ、キーボード、および／または様々な記憶装置を備えることができる。

【００２３】

好適な実施形態では、制御装置はマイクロコンピュータであり、後述するように新規のシステムソフトウェアアプリケーションを組み込む。システムソフトウェアは、Ｉ／Ｏ装置７４、例えば磁気または光ディスクドライブ、ディスケット、またはテープから、メモリ７１にロードされる１つまたはそれ以上のソフトウェアアプリケーションまたはモジュールによって表される。代替的に、システムソフトウェアは、例えば読取り専用メモリ（ＲＯＭ）７３および類似物内に格納されたファームウェアとして実現することができる。そういうものとして、本発明のクラスタツール制御ソフトウェアは、１つまたはそれ以上のコンピュータ読取り可能な媒体に格納することができる。最後に、ソフトウェアアプリケーショ

ンをひとたびロードすると、プロセッサ７２はメモリ内のこの新規のソフトウェアを実行して、クラスタツールシーケンサを実現する。一般的に、プロセッサ７２は任意の市販のマイクロプロセッサの１つ、例えばモトローラ社によって製造された６８０ｘ０型とすることができる。

【００２４】

ここでソフトウェアプロセスとして論じるプロセス工程の幾つかは、ハードウェア内に、例えばマイクロプロセッサと協働して様々なプロセス工程を実行する回路機構として実現できることが企図される。制御装置は、様々なスケジューリングルーチンを実行するようにプログラムされた汎用コンピュータとして描かれているが、ソフトウェアによって実現されるプロセスは、アプリケーション特定的な集積回路（ＡＳＩＣ）または離散回路コンポーネントなどのハードウェアとして実現することができる。そういうものとして、ここで記載するプロセス工程は、ソフトウェア、ハードウェア、またはそれらの任意の組合せによって同等に実行されるものとして、幅広く解釈されることを意図している。

【００２５】

シーケンサソフトウェアはマルチタスキングであり、定期的な呼出し信号を生成するタイマから割込み方式で呼び出される実時間タスク指向モジュールに分割される。図２に示すように、基本シーケンサソフトウェアは、シーケンサタスクモジュール２００、スクリーンタスクモジュール２０２、低速タスクモジュール２０４、チャンバタスクモジュール２０６、およびバッファチャンバタスクモジュール２０８を含む。各々のモジュール２００から２０８は呼び出されたとき、より低速のモジュールの処理に割込み、それらの処理を完了し、次いで割り込まれたルーチンの最後の出口点に戻る。この従来のマルチタスク階層動作により、システム内の他のタスクに処理に重大な影響を与えることなく、より重要な動作を実時間で処理することができる。前述のモジュール２００から２０８は各々、他のモジュール向けのメッセージまたはデータをその出口に残すことにより他のモジュールが利用するデータ構造を変化させることによって、またはその処理中に、後で命令されたモジュールの処理中に実行されるコマンドを別のモジュールに発行することによって、他のモジュールのどれとでもやりとりを行うことがで

きる。

【0026】

一般的に、シーケンサタスクモジュール２００は、プロセスレシピプログラムおよび洗浄レシピプログラムを含むウェハ製造の順序付けまたは工程毎のプロセスを制御する。モジュール２００は、ロードロック２４および２６に積載されたウェハおよびそれらのプロセスおよび洗浄レシピ、つまり様々なチャンバのプロセスのシーケンスおよび化学的処理を記載したデータ構造、つまりウェハ順序リスト（ＷＯＬ）に基づいて作動する。プロセスおよび洗浄レシピは各々、情報へのポインタによって、別のデータ構造つまりウェハ管理データ構造（ＷＭＤＳ）を介して、ＷＯＬの特定のウェハに関連付けられる。ＷＯＬおよびＷＭＳの情報に基づいて作動するシーケンサタスクモジュール２００は、特定のチャンバのための特定のプロセスレシピまたは洗浄レシピのプロセス工程を正しい時間に実行するようにチャンバタスクモジュールにコマンドを発行する。ウェハは、特定のチャンバの化学的処理に合致するようにシステムの移送動作を記載するデータ構造、つまりウェハ移動待ち行列（ＷＭＱ）に基づいて作動するシーケンサタスクモジュール２００によって、様々なチャンバを移動する。ＷＭＱにウェハがある場合には、シーケンサモジュールは、チャンバの入口スリットバルブおよびウェハ移送機構８２の制御によって移動を実行するように、バッファタスクモジュール２０８に命令する。

【0027】

チャンバタスクモジュール２０６は、真空弁の実際の制御、様々な化学物質のイオン化プロセス、ＲＦ電源の起動等を含め、ウェハ上の集積回路プロセスのための様々なレシピおよび洗浄レシピを取り扱う。チャンバタスクモジュール２０６は、データ構造つまりチャンバデータ構造（ＣＤＳ）に基づくその動作によって、特殊化された処理チャンバで起こるプロセスを監視し制御する。

【0028】

スクリーンタスクモジュール２０２は、オペレータが特定のデータを他のモジュールのデータ構造内にプログラムし、対話的な方法でタスクモジュール（またはその他のプログラムオブジェクト）に、それらが実行される前およびそれらの

実行中に、メッセージまたはコマンドを残すことを可能にする。スクリーンタスクモジュール２０２の主要部の１つは対話型テキストエディタであり、オペレータがオペレータインタフェースまたはディスプレイを介してデータおよびコマンドを入力したり編集し、それらをプロセスのための制御および管理モジュール（図示せず）に格納することを可能にする。スクリーンタスクモジュール２０２はまた、システム１０の自動制御シーケンスの実行中に、オペレータが様々な状態、アラーム、および監視情報を表示することを可能にする。

【００２９】

低速タスクモジュール２０４もまた、比較的低速で行われるシステムの特定のタスクを監視し、かつタイミングを取るために設ける。例えば、チャンバの洗浄時間は、特定のチャンバのＲＦコイルが付勢されていた、またはアクティブであった時間の量を決定することによって監視することができる。

【００３０】

ここで、プロセス制御によって使用されるデータ構造を、図３から６に関連してより詳細に説明する。ＷＯＬデータ構造は、図３により詳細に開示されている。ウェハ順序リストは複数の情報ブロック１、２．．．ｎを備えており、各々に、ウェハ加工システム１０により加工するように現在計画されているウェハの１つが記載されている。各情報ブロック、例えば２１０と表示されたブロック１は、ウェハを一意に識別するウェハ識別番号およびそれが属するロット番号を含む。情報ブロック２１０はさらに、ウェハの加工を制御しウェハ順序リストの情報ブロック２１０に対応するウェハ管理ブロック１、２、．．．ｎを有するＷＭＤ（図６）へのポインタをさらに含む。これらのフィールドの後に２つのフラッグフィールドがある。１つは、ウェハをデッドロックされたプロセスウェハとして識別するためのもの、もう１つは、それをデッドロックされ配向された（事前配向）ウェハとして識別するためのものである。情報ブロック２１０の最後のフィールドは、ウェハの加工スケジュールを組むことができる処理チャンバのグループを識別するグループチャンバマスクである。ＷＯＬの残りの情報ブロック２、３．．．ｎは情報ブロック２１０と同様に構成され、クラスタツールで加工される残りのウェハについての情報を含む。

【００３１】

ウェハ管理データ構造は図６により詳しく開示されており、構造的にはウェハ順序リストに類似している。ＷＭＤはウェハ管理ブロック１、２、．．．ｎを有し、各々がそれぞれのＷＯＬ情報ブロックに対応し、したがって特定のウェハに対応する。ウェハ管理ブロック、例えば２１６と表示されたブロック１のフィールドは、ウェハのロット番号およびウェハ識別番号を含む。それに加えて、状態フィールドはプロセス状態を記載する。さらに、ウェハのプロセスレシピおよびそれに対応付けられた洗浄レシピを記載する、プログラムシーケンス識別がある。ウェハ管理ブロック２１６の最後のフィールドは、ウェハがそのプロセスを開始する供給源のカセット番号およびスロットを記録する。理想的には、そのような供給源スロットは、加工後にウェハが戻されるスロットである。ＷＭＤＳの残りのウェハ管理ブロック２、３．．．ｎは、ウェハ管理ブロック２１６と同様に構成され、システム１０内の残りのウェハについての情報を含む。

【００３２】

ＷＭＱデータ構造を図４により詳しく開示する。ウェハ移動待ち行列は移動を必要とするウェハのリストであり、移動すべきウェハおよびその移動経路（供給源および行先）を識別するフィールドを含む多数の識別子ブロック１、２、．．．ｎから成る。２１２と表示された識別子ブロック１は、移動すべきウェハを識別するウェハ順序リスト内のウェハ情報ブロックに対するポインタを記憶するための１つ、および供給源および行先処理チャンバを、あるいは供給源または行先がカセットまたはロードロックの場合はエレベータまたはカセットのスロットを識別するための別の２つのフィールドを含むフィールドを有する。ＷＭＱデータ構造の残りの識別子ブロック２、３．．．ｎは、識別子ブロック２１２と同様に構成され、移動が必要な他のウェハの同様のデータを含む。

【００３３】

チャンバデータ構造を図５により詳しく開示する。チャンバデータ構造は、処理チャンバおよびそれらの状態のリストである。チャンバデータ構造は多数の状態ブロック１、２、．．．ｎから成り、２１４と表示された状態ブロック１は、一意の数字によるチャンバと、そのチャンバで実行中のレシピ工程の残りの時間

を識別するフィールドを含む。別のフィールドは、チャンバが使用中か空いているかを示すフラッグとして使用され、セットされたときは、チャンバが使用中であってプロセスレシピ工程または洗浄レシピ工程を実行していることを示す。最後のフィールドは、特定のチャンバがその最後の洗浄以来どれだけの数のウェハを処理してきたか、ウェハ数を格納するために使用される。処理チャンバデータ構造の残りの状態ブロック２、３．．．ｎは、状態ブロック２１４と同様に構成され、システム内の他のチャンバに関する同様のデータを含む。

【００３４】

ここで、図７に関連して、本発明に従って構成したプレシーケンサを含むシーケンサタスクモジュール２００について、さらに詳しく説明する。シーケンサタスクモジュールの主プログラム２１８は、５０ｍｓごとに実行されるルーチンのリストである。一般的に、タスクモジュール２００はブロック２２０で、他のルーチンからの入力メッセージ、データ、またはコマンドを処理して、次のタスクを実行するために必要となる他のルーチンからの情報を更新することによって開始する。

【００３５】

ブロック２２２で、モジュール２００は、プロセスの隘路、主としてウェハのデッドロックを低減または除去するために、先を見通して上述の処理チャンバ１６から３０間のウェハの移動スケジュールを組み、優先順位を決定する動的先読みルーチンまたはプレシーケンサを実行する。この事前順序付けにより、システム１０はチャンバをより効果的に利用し、クラスタツールのスループットを高めることが可能になる。

【００３６】

動的先読みプログラム２２２は、シーケンサタスクモジュール２００が加工のシーケンスを順序付けるために使用するウェハ順序リストを調べ、シーケンス内のどのウェハが将来スループット上の問題を引き起こすかを決定する。シーケンス内の識別された問題ウェハのリストから、プログラムはウェハ順序リスト内のウェハ識別データおよびウェハ移動待ち行列の順序を変更して、これらの問題を緩和または除去する。これらの問題を低減するためのウェハの移動は、どのタイ

プのウェハの移送がスループットを最も改善するかによって優先順位を決定され、それ以後の加工を行う前に必要な修正が行われる。プレシーケンサルーチン２２２は５０ｍｓごとに実行されるので、ひとたび問題が識別されると、それは瞬時に修正される。

【００３７】

ＷＭＱおよびＷＯＬまたはその他の必要な情報が更新された後、シーケンサタスクモジュール２００は、ＷＭＱによって画定される通り予め定められたシーケンスでウェハを移動するように、ブロック２２４によって表わされるバッファチャンバタスクモジュールにコマンドを発行し、ウェハがそれらの適切なチャンバに移動した後、ウェハのプロセスレシピまたは特定のチャンバの洗浄レシピのいずれかを順序付けるように、ブロック２２６のチャンバタスクモジュールにコマンドが発行される。そういうものとして、チャンバタスクがブロック２２８で始動する。ひとたびチャンバタスクが始動すると、タスクシーケンサモジュール２００はブロック２２９で休眠し、次の目覚まし呼出しを待つ。

【００３８】

動的先読みルーチン２２２のより詳細な機能流れ図を、図８ａから８ｇに開示する。ブロック２２２によって表すプレシーケンサプログラムの一般的機能は図８ａに詳述されており、そこでのブロック２３０の第１機能は、ウェハ順序リストの先を見通して、シーケンス中間デッドロックウェハを探すことである。これらは、それらの加工の一部が終了して、次の処理チャンバに移動する準備ができているが、次の処理チャンバが使用中であることが判明したウェハである。しかし、それらはまた、それらが現在配置されている処理チャンバまたは資源を開放しないことによってシステムを停滞させてもいる。これは、これおよびその他のウェハの可能な加工時間の実質的な部分が、そのウェハの行先チャンバがデッドロックウェハを受け入れることができるように（利用可能に）なるまで停止する、デッドロックの可能性を生じる。この問題の解決策は、デッドロックしたウェハをその供給源処理チャンバから取り出して、そのウェハの行先チャンバが利用可能になるまで、それをシステム１０の待機位置に、好ましくはロードロックチャンバに戻すことである。これにより、デッドロックウェハが行先チャンバを待

っている間、開放された供給源チャンバで他のウェハを処理することが可能になる。ロードロックは、待機位置の実例である。デッドロックウェハは使用しないプロセスチャンバ、あるいは使用しうる特別な待機位置チャンバに一時的に配置しうる。待機位置チャンバは、処理チャンバを待っている複数のデッドロックウェハを格納するためのウェハエレベータを含むことができる。

【００３９】

ブロック２３２の次の機能は、ウェハ順序リストの先を見通して、配向チャンバ３０内にデッドロックした配向済みのウェハが無いか調べることである。これらは、配向が終了し、次の処理チャンバに移動する準備ができているが、次の処理チャンバが使用中であることが判明したウェハである。しかし、それらはまた、デッドロックした配向済みウェハを含む配向チャンバを他のウェハの処理のために開放しないことによって、クラスタツールを停滞してもいる。これは、行先チャンバが配向済みウェハを受容する準備ができるようになるまで、これおよびその他のウェハの可能な加工の多くが停止する、デッドロックの可能性を生じる。この問題の解決策は再び、デッドロックしたウェハを配向チャンバ３０から取り出して、行先チャンバが使用できるようになるまで、それをシステム１０の待機位置、好ましくはロードロックに戻すことである。これにより、デッドロックした配向済みウェハが行先チャンバへの移動を待っている間に、開放された配向チャンバで他のウェハを配向することが可能になる。ブロック２３４および２３６における次の２つの機能は同様である。それらはＷＯＬの先を見通して、待機位置またはロードロックに戻されたプロセスウェハおよび戻された配向済みウェハが無いかを調べ、将来の加工のためにそれらのスケジュールを組み直す。

【００４０】

ひとたび問題ウェハ（デッドロック態になりやすいもの）の識別およびスケジュールの組み直しが達成されると、（ａ）デッドロックしたウェハをロードロックに戻し、（ｂ）解放された処理チャンバまたは配向チャンバに他のウェハを移動し、かつ（ｃ）スケジュールを組み直したロードロックウェハを含むその他のスケジュールされたウェハを移動するためのウェハ移送機構８２の運動は、ブロック２３８でウェハの種類を試験することによって、優先順位を付けることがで

きる。ウェハの種類は、デッドロックしたウェハ、配向したウェハ、シーケンス中間のウェハおよび類似物を含む。試験は、どの種類のウェハがウェハ移動待ち行列にリストされているかを明らかにし、次いでウェハ識別データが、移動すべきウェハの種類に基づいて再編成される。例えば、デッドロックしたシーケンス中間のウェハは、配向済みであるが未加工のウェハより高い優先順位をもつ。さらに、配向済みであるが未加工のウェハは、配向されていないがそれでもロードロックから取り出さなければならないウェハより高い優先順位を持つ。ウェハ移動待ち行列に優先順位を付けた後、待ち行列内のウェハは、ブロック２４０に示す通り、それらのそれぞれの行先に移動する。

【００４１】

同定先読みによりデッドロックされたシーケンス中間のウェハを識別するために図８ａのブロック２３０を実現する詳細ルーチンを、今から図８ｂに関連してより詳しく説明する。ウェハ順序リストにある全てのウェハは、主ルーチンに戻る前に図８ｂのルーチンによって検査されるが、明確を期すために１つのウェハのみの工程を記載する。ルーチンはブロック２４２で最初に、現在ウェハを含んでいるチャンバ（供給源チャンバ）のチャンバ識別を読み取り、それがグループチャンバであるかどうか、つまり次の工程でウェハを受け取ることができる多数の行先チャンバがあるかどうかを決定する。チャンバがグループチャンバでなければ、ルーチンは見つかったその特定のチャンバＩＤ番号を使用し、ブロック２４６に進む。チャンバ識別が、そのウェハがグループチャンバ内に含まれることを示した場合には、そのグループの各チャンバのチャンバデータ構造を調べることにより、ブロック２４４でレシピの残りの時間を読み取ることによって、グループの中で次に利用可能なチャンバを選択する。ブロック２４６内でプログラムは供給源チャンバの識別および行先チャンバの識別を続け、行先チャンバが次の移動に利用可能であるかどうかを決定する。チャンバデータ構造の使用中状態フラッグおよびレシピフィールドの残り時間により、行先チャンバの利用可能性を試験する。

【００４２】

チャンバが利用可能である場合には、ブロック２５２でウェハ移動待ち行列デ

ータ構造にウェハ識別データを入力し、これが調べるべきウェハ順序リストの最後のエントリであれば、プログラムは戻る。チャンバが利用できない場合には、ブロック２５０で、その特定のウェハにデッドロックフラッグがセットされ、今使用中である次の処理チャンバの識別が、ウェハ順序リストに保存される。そのウェハの行先チャンバフィールドは次いでロードロックに変更され、ウェハ識別データが、ブロック２５２でウェハ移動待ち行列に入力される。そういうものとして、バッファチャンバタスクモジュールの実行後に、ウェハはロードロックに移動する。ＷＯＬの全てのウェハが試験されると、ルーチンはその後、主タスクモジュールに戻る。

【００４３】

動的先読みによりデッドロックされた配向済みウェハを識別するために図８ａのブロック２３２を実現する詳細なルーチンを、図８ｃに関連してより詳しく説明する。ウェハ順序リストの全てのウェハは、主ルーチンに戻る前に図８ｃのルーチンによって検査されるが、明確を期すために、１つのウェハのみの工程を記載する。ルーチンはブロック２５４で最初に、現在ウェハを含んでいるチャンバ（供給源チャンバ）のチャンバ識別を読み取り、それがグループチャンバであるかどうか、つまり次の工程でウェハを受け取ることができる多数の行先チャンバがあるかどうかを決定する。チャンバがグループチャンバでなければ、ルーチンは見つかったその特定のチャンバＩＤ番号を使用し、ブロック２５８に進む。チャンバ識別が、そのウェハがグループチャンバ内に含まれることを示した場合には、そのグループの各チャンバのチャンバデータ構造を調べることにより、ブロック２５６でレシピの残りの時間を読み取ることによって、グループの中で次に利用可能なチャンバを選択する。ブロック２５８内でプログラムは供給源チャンバの識別および行先チャンバの識別を続け、行先チャンバが次のスケジュールされた移動に利用可能であるかどうかを決定する。チャンバデータ構造の使用中状態フラッグおよびレシピ残り時間フィールドにより、行先チャンバの利用可能性を試験する。

【００４４】

ブロック２６０で試験して次のチャンバが利用可能である場合には、ブロック

２６４でウェハ移動待ち行列データ構造にウェハ識別データを入力し、これが調べるべきウェハ順序リストの最後のエントリであれば、プログラムは戻る。チャンバが利用できない場合には、ブロック２６２で、その特定のウェハに事前配向フラッグがセットされ、今使用中である次の処理チャンバの識別が、ウェハ順序リストに保存される。そのウェハの行先チャンバフィールドは次いでロードロックに変更され、ウェハ識別データが、ブロック２６４でウェハ移動待ち行列に入力される。そういうものとして、バッファチャンバタスクモジュールの実行後に、ウェハはロードロックに移動する。ＷＯＬの全てのウェハが試験されると、ルーチンはその後、図８ａのブロック２３２に戻る。

【００４５】

シーケンス中間ウェハおよび配向済みウェハについて処理した後、ルーチンは図８ｄに移り、ここでルーチンは、戻されたプロセスウェハおよび戻された配向済みウェハが無いかウェハ順序リストを検査し、当初の行先チャンバに向けてそれらのスケジュールを組み直す。図８ｄのルーチンはしたがって、図８ａのブロック２３４および２３６によって表わされた方法の工程を実現する。ルーチンは、ウェハ順序リスト内の全てのウェハについて、ステップ２６６から２７８のグループを実行し、ブロック２６６で、ウェハ順序リストで現在アドレス指定されている情報ブロックが第１ウェハを識別するかどうかを試験することによって開始する。次に、ブロック２６８で、ルーチンは、現在アドレス指定されているウェハが位置するチャンバの識別を決定する。チャンバＩＤは次にブロック２７０で試験され、それがロードロックチャンバであるかどうかが決定される。そうでない場合には、プログラムは、ブロック２７８の試験に移ることによって、次のウェハに移動する。しかし、ウェハがロードロックにある場合には、それは戻されたウェハであるかもしれない。ブロック２７２でシーケンス中間フラッグが試験され、ブロック２７４で配向済みフラッグが試験され、いずれかがセットされているかどうかが決定される。いずれかのフラッグがセットされている場合には、これは、スケジュールの組み直しを待っている戻されたウェハである。次のプロセスは図８ｅに移り、これは、ブロック２７６によって呼び出されたスケジュール組み直しルーチンを実現する。ブロック２７２、２７４の試験で、デッドロ

ックフラッグがセットされていないことが示された場合には、プログラムはブロック２７８、２８０に戻り、そこでウェハ順序リストの次の情報ブロックが検査される。ウェハ順序リストの全てのウェハが検査された後、そのときプログラムは図８ａのブロック２３４、あるいは２３６に戻る。

【００４６】

図８ｅは、図８ｄのブロック２７６のウェハスケジュール組み直しプロセスを実現するルーチンの流れ図を示す。ルーチンはブロック２８２で最初に、現在ウェハを含んでいるチャンバ（供給源チャンバ）のチャンバ識別を読み取り、それがグループチャンバであるかどうか、つまり次のステップでウェハを受け取ることができる多数の行先チャンバがあるかどうかを決定する。グループチャンバの識別情報はグループマスクに含まれる。チャンバがグループチャンバでなければ、ルーチンは見つかったその特定のチャンバＩＤ番号を使用し、ブロック２８６に進む。チャンバ識別が、そのウェハがグループチャンバ内に含まれることを示した場合（つまり、そのチャンバがグループマスクに入っている場合）には、ブロック２８４で、そのチャンバグループマスクおよび各チャンバのチャンバデータ構造を調べることにより、レシピの残り時間を読み取ることによって、グループの中で次に利用可能なチャンバを選択する。ブロック２８６で、ルーチンはチャンバデータ構造を検査して、行先チャンバ（ターゲット）が利用可能であるかどうかを決定する。ブロック２８８で、ルーチンは、ターゲットチャンバが利用可能であるかどうかを照会する。照会が否定的に応答された場合には、ルーチンは図８ｄのブロック２７６に戻る。従って、ウェハはスケジュールを組み直されず、それはもっと長く待たなければならない。ブロック２８８の照会が好適的に応答された場合には、ルーチンはブロック２９０に入り、ＷＭＱデータ構造を更新する。そういうものとして、次のバッファチャンバタスクの実行後に、ウェハはロードロックの待機位置から加工用のチャンバに移動する。次にルーチンは図８ｅのブロック２７６に戻る。

【００４７】

デッドロックされたシーケンス中間処理ウェハ、デッドロックされた配向済みウェハを識別して、その移動のスケジュールを組み直し、かつ戻されたシーケン

ス中間ウェハおよび事前配向されたウェハを識別してその移動のスケジュールを組み直した後、プログラムはウェハ移動待ち行列の優先順位を決定する。このプロセスのステップの詳細な機能流れ図を、図８ａのブロック２３８を実現する図８ｆにさらに詳しく記述する。ウェハ移動待ち行列は、プロセスおよび洗浄レシピ、プレシーケンサ、スクリーンタスクモジュール（オペレータ制御）および類似物をはじめ、ウェハを移動する必要のある様々なプログラムから無作為の順序でロードされる。図８ｆのルーチンはこれらの移動の順序付けをやり直し、プロセスで発生し得る隘路に優先順位を付けて、その閉塞を防止する。ウェハ順序リストの順序付けをやり直して、優先順位の第一に洗浄ウェハを、第二に加工済みウェハを、第三にデッドロックされたシーケンス中間および配向済みウェハを、第四にロードロック内の戻されたシーケンス中間および配向済みのウェハを、そして第五にロードロック内の未加工ウェハを置く。この方法により、洗浄を必要とする処理チャンバが停止されないので、そのチャンバは最短可能な時間で利用可能になる。さらに、完了したプロセスウェハはシステムから取り出すか、または次のステップに移動することができ、それによって資源を第二優先順位で移動すべき他のウェハのために解放することができる。デッドロックされたシーケンス中間または配向済みのウェハは第三優先順位で移動し、特定の処理チャンバまたは配向チャンバをそれぞれ解放する。次いで、戻された配向済みウェハおよび戻されたシーケンス中間ウェハは、起こり得るスケジュールの合間を利用して移動し、これらのウェハにおける加工を完了する。最後に、ロードロック内の未加工ウェハは移動待ち行列の最後尾である。それにより、新しいウェハは、他の部分的に加工されたウェハが処理されるまで、新しいスケジュールを作成しない。この方法により、ウェハの最小限の移動が企図され、通し加工効率が著しく向上する。

【００４８】

図８ｇのプログラムは事実上、ウェハを移動するようにバッファチャンバタスクモジュールにコマンドを生成し、そういうものとして、図８ａのブロック２４０を実現する。ＷＭＱの順序付けをやり直した後、リストのウェハの移動が開始される。ウェハ移動待ち行列の各エントリはブロック３０２で順次アドレス指定

され、現在の関心対象の特定のウェハを移動するように、バッファタスクモジュールにコマンドが出される。リストのウェハに対するウェハ移動コマンドが出された後、ルーチンはブロック３０４で、ウェハ順序リストのフラッグでウェハ識別データを調べることにより、そのウェハが戻されたシーケンス中間または配向済みのウェハであるかどうか、かつ、そのウェハの供給源チャンバがロードロックであるかどうかを決定する。両方の条件が満たされる場合には、それがどの種類のウェハと決定されたか（つまりウェハがデッドロックされたウェハか、それとも事前配向されたウェハか）によって、事前配向フラッグまたはデッドロックフラッグが、ブロック３０６でウェハ順序リストにリセットされる。これらのフラッグをクリアした後、プログラムはブロック３０８で、これがウェハ移動待ち行列の最後のウェハであるかどうかを決定する。最後のウェハでなければ、プログラムはブロック３０２に戻り、別のウェハ移動コマンドを発行する。最後のウェハコマンドが発行された後、プログラムは図８ａのブロック２４０に戻る。

【００４９】

図９ａから９ｂの機能流れ図でさらに詳述するバッファチャンバタスクモジュール２０８は、ウェハ移送機構８０を制御して、処理チャンバとロードロックとの間のウェハの移送を実行するプログラムである。バッファチャンバタスクモジュール２０８は２０ｍｓごとに割込み方式で呼び出され、バッファチャンバに関連するその他のタスクと一緒にこの動作を実行する。バッファチャンバタスクモジュール２０８がブロック３１０で、特定のウェハを移動するコマンドを持っていると決定した場合、それは、そのタスクを達成するために、ウェハ移動プログラム３１２を呼び出す。ウェハ移動プログラム３１２は前述の通り、移送機構８２がウェハをその供給源チャンバから持ち上げ、それをその行先チャンバに移動するのを制御する。このプログラムはまたチャンバの入口スリット弁の開閉を制御して、必要に応じて移送を可能にし、加工中および洗浄中の分離を達成する。プレシーケンサの別の特徴に従って、プログラムはウェハ移動コマンドを受け取った後、行先がカセット内のスロットであるかどうかをも試験する。この試験が真である場合には、プレシーケンサはウェハ移動待ち行列の先を見通して、ロードロックに結び付けられた別のウェハがあるかどうかを決定し、ブロック３１６

でウェハの交換後にロードロックのスリット弁を開けたままにする。そういうものとして、交換のたびにロードロックのスリット弁を開閉することなく、複数のウェハ交換を達成することができる。その後、ルーチンはブロック３１７に進み、そこでその他のタスクを完了した後、その割込み出口点に戻る。

【００５０】

図９ｂのルーチンの詳細流れ図は、図９ａのステップ３１６を実現する。最初にプログラムはブロック３２０で、ウェハ移動待ち行列にもっとウェハがあるかどうかを決定し、もし無ければ、否定的標識をウェハ移動プログラム３１２に返す。もしあれば、次のウェハがブロック３２２、３２４で試験され、その供給源カセット識別が、返ってくるウェハカセット識別に等しいかどうかが調べられる。試験が好適的に応答された場合、ルーチンはブロック３２６に続き、そこで次のウェハの現在の位置を決定する。ウェハの現在の位置を決定した後、プログラムはブロック３２８で、その位置が返ってくるカセット識別と等しいかどうかを試験する。イエスの場合、プログラムはウェハ移動プログラムに肯定的な応答を返す。そうでない場合、プログラムはブロック３２０に逆戻りし、そこでウェハ移動待ち行列を試験して、もっとウェハがあるかどうかを調べる。これにより、ロードロックに配置されておりロードロックから取り出すようにスケジュールされたウェハがウェハ移動リストにあることを示すフラッグが、ウェハ移動プログラムに生成され、現在のウェハはロードロックに移動するようにスケジュールされる。スリット弁ドアは、入ってくるウェハをロードロックに移動するために開かなければならず、その後、ウェハ移動プログラムが、出て行くウェアを移送するために後で再び開くためにだけ閉じる必要が無いように、意図的に開けたままにする。ロードロックではかなりのウェハ移送量があるので、特に事前順序付けと共に、この洗練された時間節約は、ロードロック機構の効率を高め、かつ摩耗を低下する。

【００５１】

本発明を好適な実施形態に関連して説明したが、この明細書は、本発明の範囲を明記した特定の形態に制限することを意図するものではなく、その反対に、請求の範囲によって画定される本発明の精神および範囲内に含まれる全ての代替例

、変形例、および均等物を網羅することを意図している。

【図面の簡単な説明】

【図１】

本発明に従って作動するマルチチャンバ加工システムの概略平面図である。

【図２】

図１に示した制御装置のプロセス制御プログラムの絵画的システム流れ図である。

【図３】

図２に示したプロセス制御プログラムによって使用されるウェハ順序リストデータ構造の表形式表現である。

【図４】

図２に示したプロセス制御プログラムによって使用されるウェハ移動待ち行列データ構造の表形式表現である。

【図５】

図２に示したプロセス制御プログラムによって使用されるチャンバデータ構造の表形式表現である。

【図６】

図２に示したプロセス制御プログラムによって使用されるウェハ管理データ構造の表形式表現である。

【図７】

図２に示したプロセス制御プログラムのシーケンサタスクモジュールの詳細流れ図である。

【図８ａ】

図７に示したシーケンサタスクモジュールの先読み機能の詳細流れ図である。

【図８ｂ】

図７に示したシーケンサタスクモジュールの先読み機能の詳細流れ図である。

【図８ｃ】

図７に示したシーケンサタスクモジュールの先読み機能の詳細流れ図である。

【図８ｄ】

図７に示したシーケンサタスクモジュールの先読み機能の詳細流れ図である。

【図８ｅ】

図７に示したシーケンサタスクモジュールの先読み機能の詳細流れ図である。

【図８ｆ】

図７に示したシーケンサタスクモジュールの先読み機能の詳細流れ図である。

【図８ｇ】

図７に示したシーケンサタスクモジュールの先読み機能の詳細流れ図である。

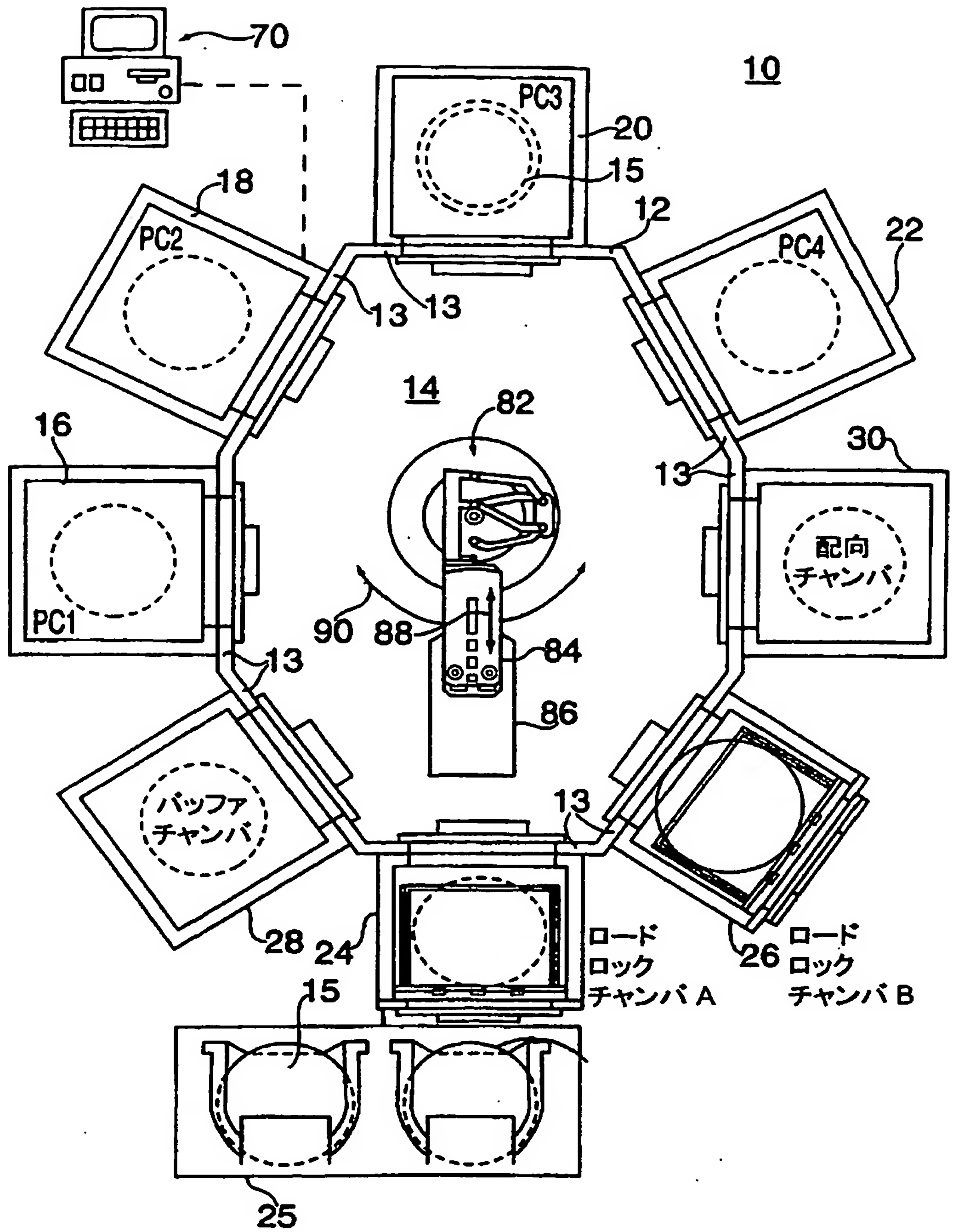
【図９ａ】

図２に示したプロセス制御プログラムのバッファチャンバタスクモジュールの詳細流れ図である。

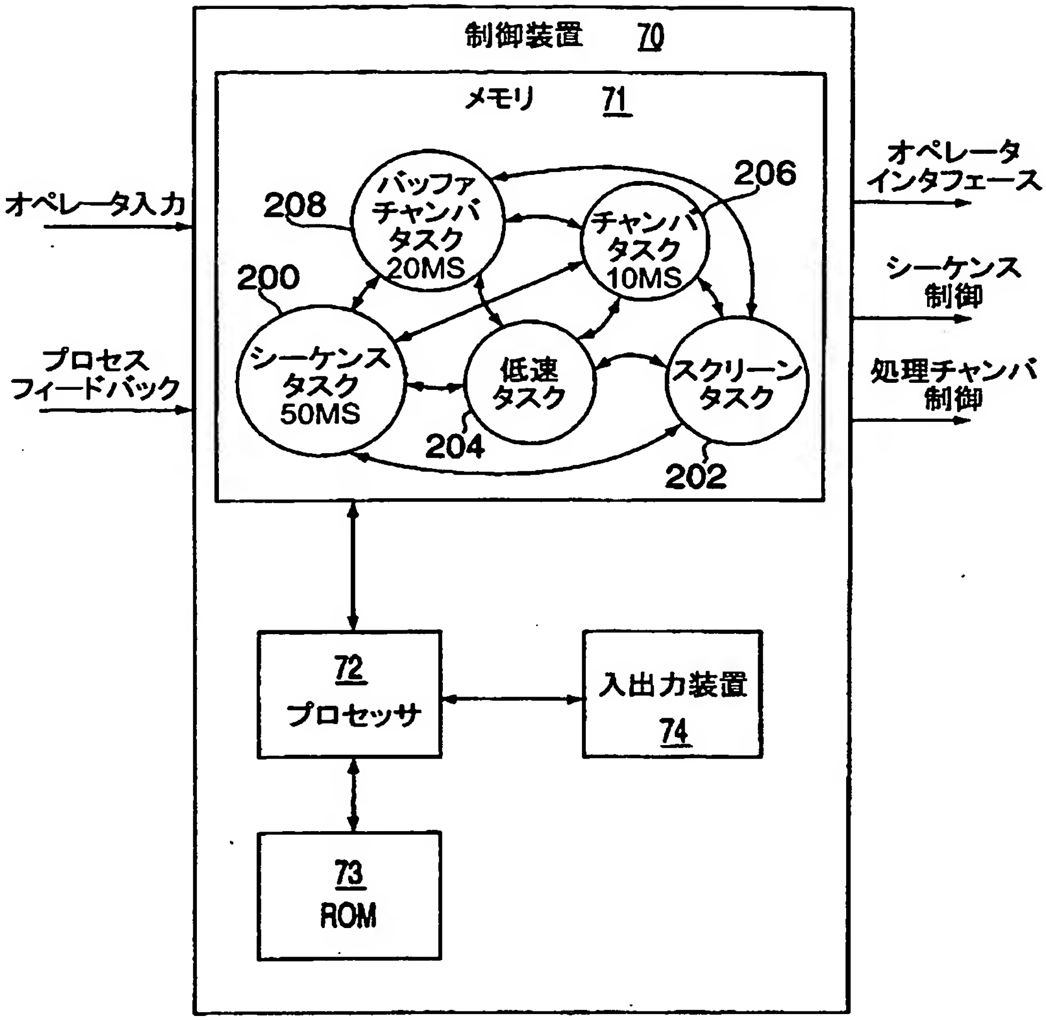
【図９ｂ】

図２に示したプロセス制御プログラムのバッファチャンバタスクモジュールの詳細流れ図である。

【図１】


70
10
PC3
20
15
12
18
PC2
13
13
PC4
22
14
82
16
30
13
配向
チャンバ
PC1
90
88
84
13
86
バッファ
チャンバ
13
28
24
ロード
ロック
チャンバＡ
26
ロード
ロック
チャンバＢ
15
25

【図２】



制御装置 70
メモリ 71
オペレータ入力
プロセス
フィードバック
オペレータ
インタフェース
シーケンス
制御
処理チャンバ
制御
バッファ
チャンバ
タスク
20MS
208
チャンバ
タスク
10MS
206
200
シーケンス
タスク
50MS
低速
タスク
204
スクリーン
タスク
202
72
プロセッサ
入出力装置
74
73
ROM

【図３】

ウェハ順序リスト

210

ID	情報
1	- ウェハのロット番号 - ウェハID - ウェハ管理ポインタ - フラッグ(事前配向、デッドロック) - グループチャンバマスク ・・・・・・・・・・・・
2	同上

【図４】

ウェハ移動待ち行列

212

ID	情報
1	- ウェハ移動リストポインタ - 供給源チャンバ/スロット - 行先チャンバ/スロット ・ ・ ・
2	同上

【図5】

チャンバデータ構造

ID	情報
1	-チャンバID -使用中フラッグ -レシピ残り時間 -洗浄、検査に使用されたウェハ数 ・・・・・
2	同上

214

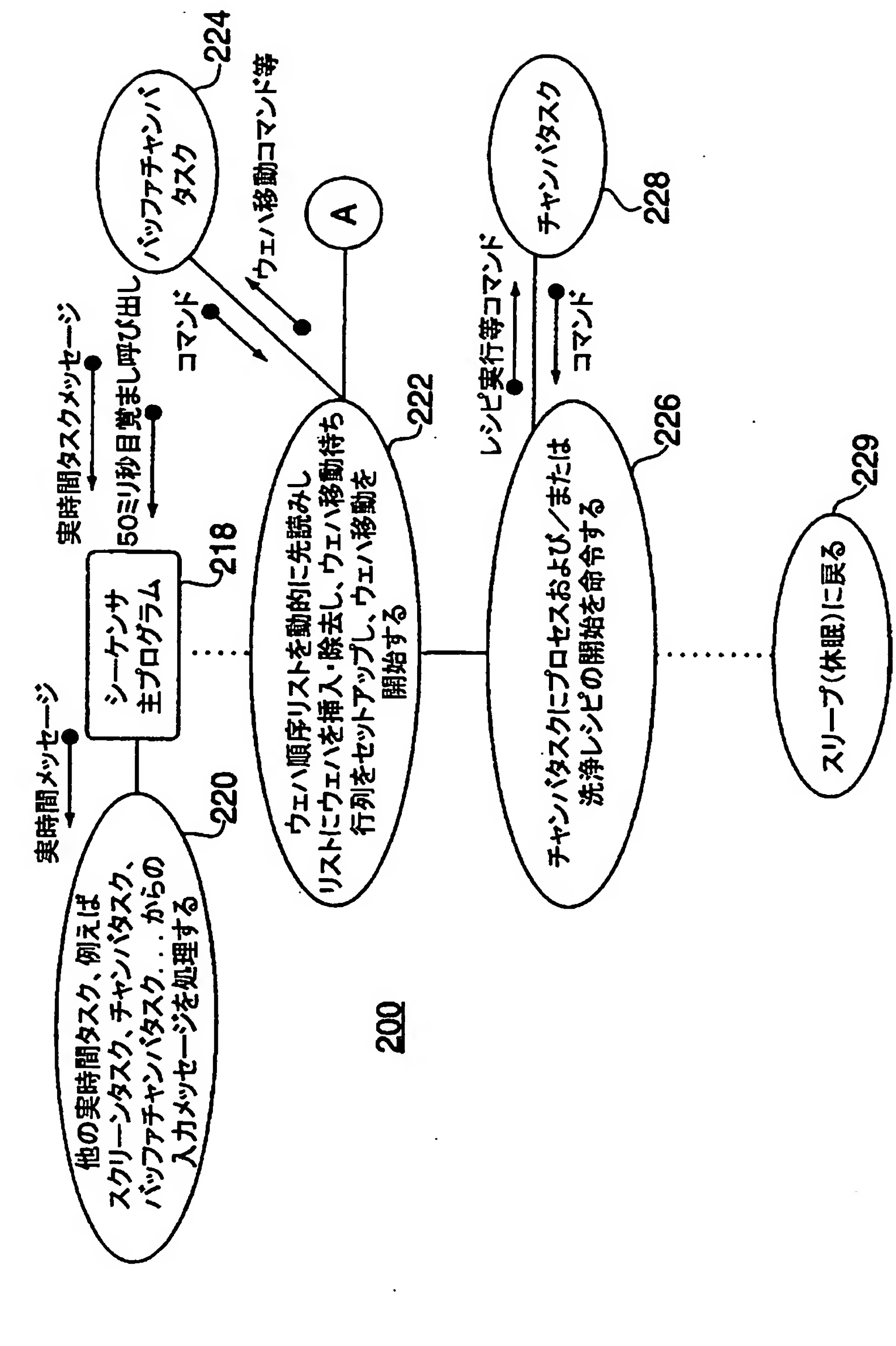
【図6】

ウェハ管理データ構造

ID	情報
1	-ウェハのロット番号 -ウェハ番号 -プログラムシーケンスID -ウェハ状態 -供給源カセット -供給源スロット ・
2	同上

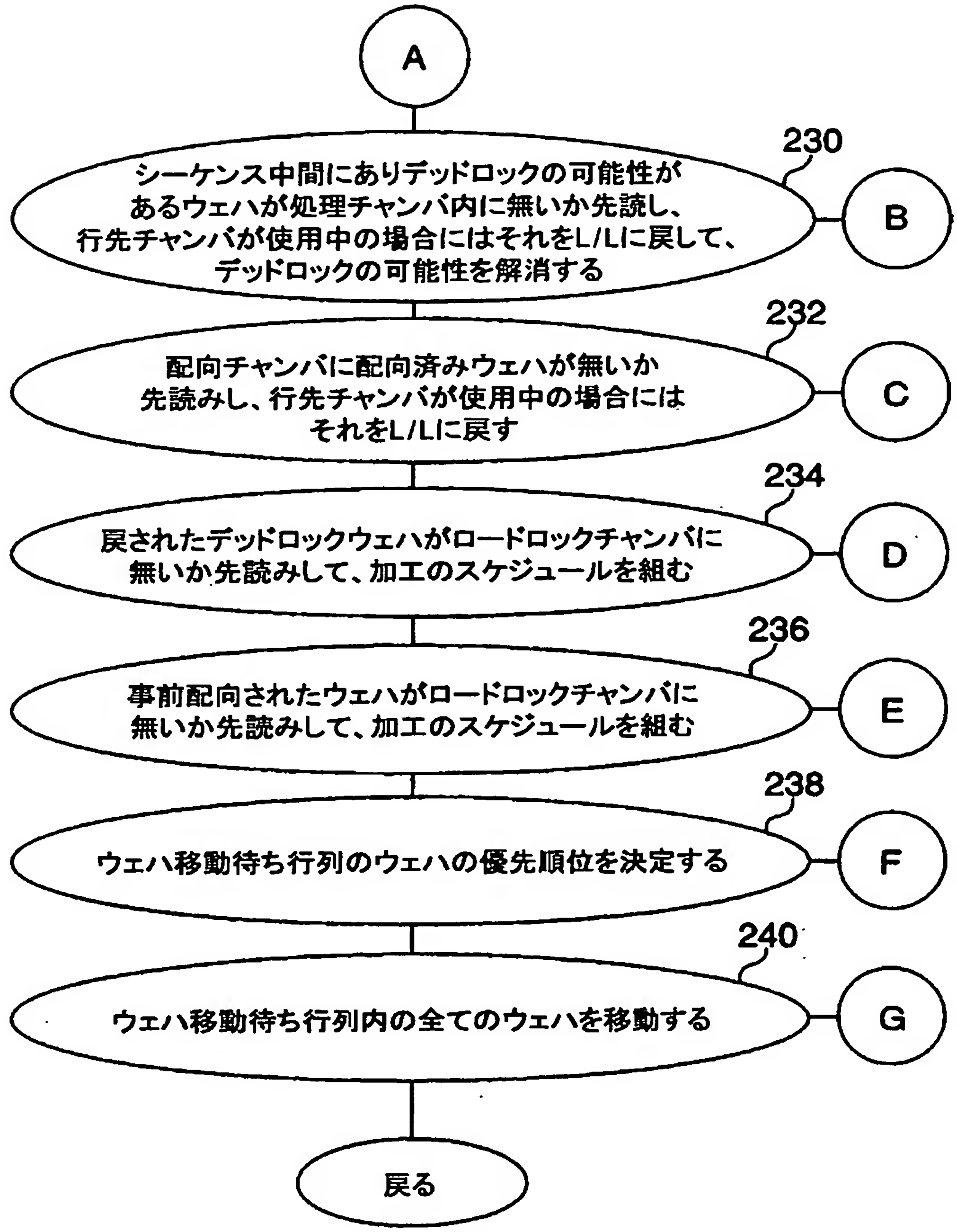
216

【図７】

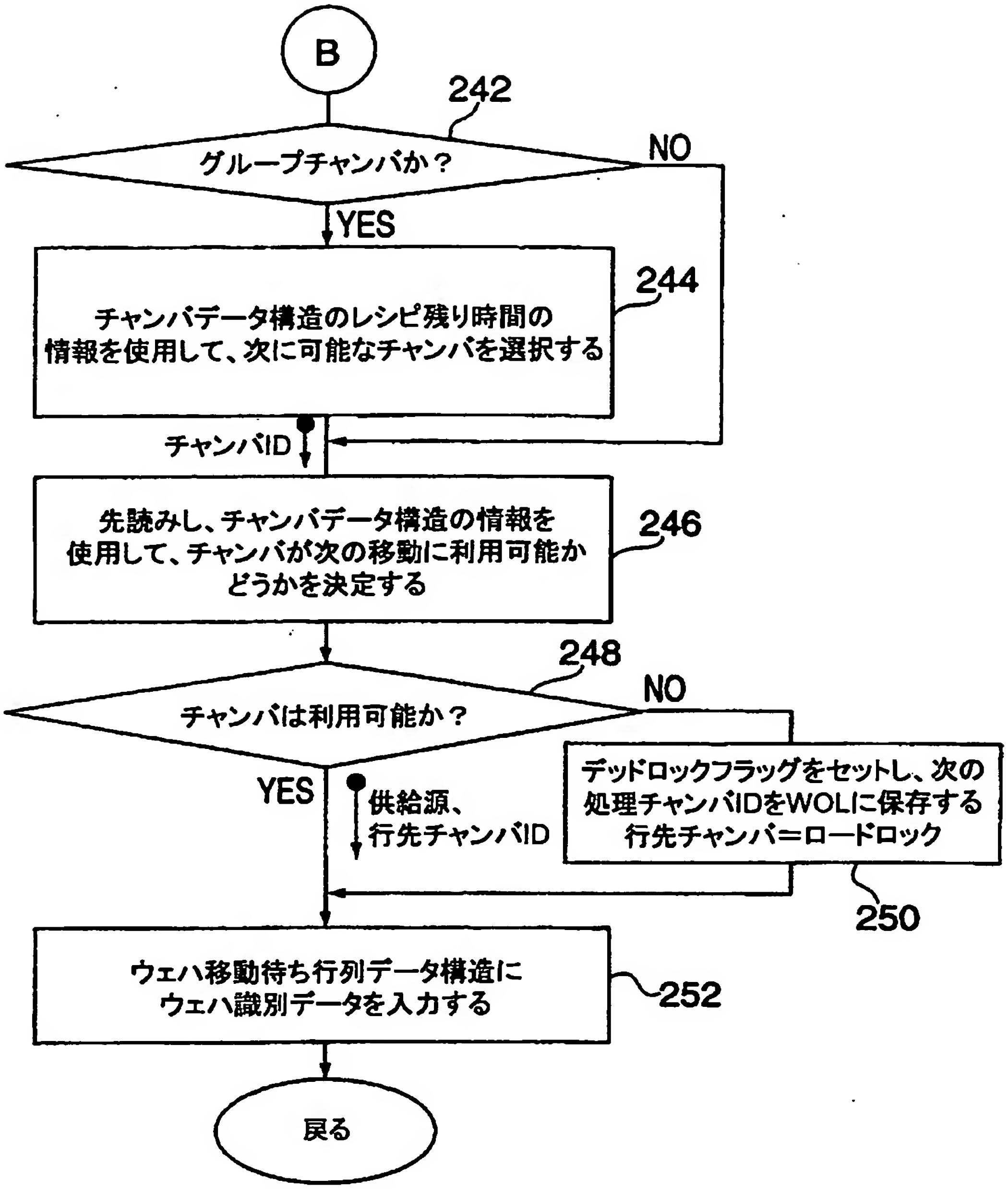


200
他の実時間タスク、例えば
スクリーンタスク、チャンバタスク、
バッファチャンバタスク．．．からの
入力メッセージを処理する
220
実時間メッセージ
シーケンサ
主プログラム
218
実時間タスクメッセージ
50ミリ秒目覚まし呼び出し
ウェハ順序リストを動的に先読みし
リストにウェハを挿入・除去し、ウェハ移動待ち
行列をセットアップし、ウェハ移動を
開始する
222
コマンド
ウェハ移動コマンド等
バッファチャンバ
タスク
224
A
チャンバタスクにプロセスおよび／または
洗浄レシピの開始を命令する
226
レシピ実行等コマンド
コマンド
チャンバタスク
228
スリープ（休眠）に戻る
229

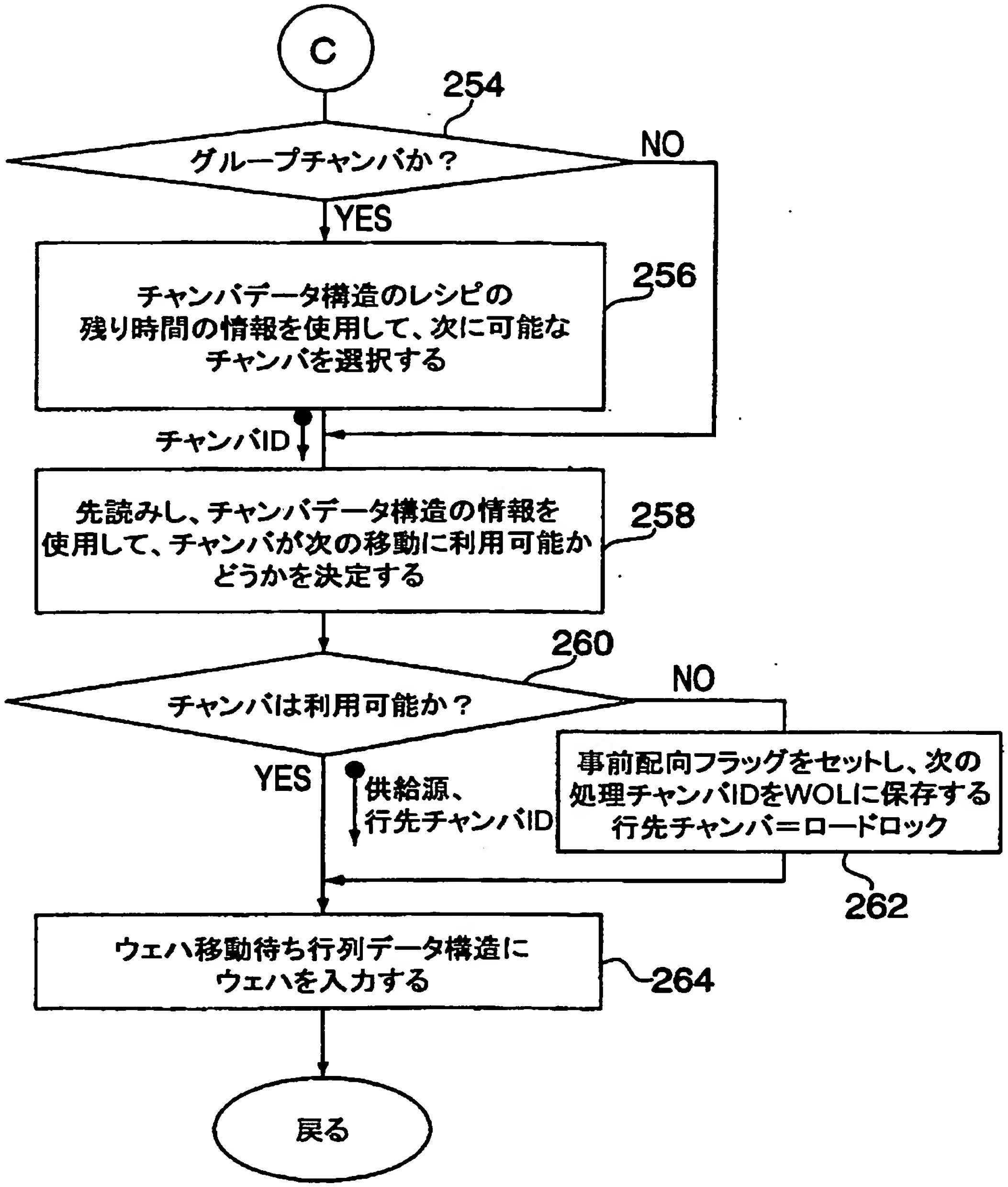
【図８ａ】


A
230
シーケンス中間にありデッドロックの可能性があるウェハが処理チャンバ内に無いか先読し、行先チャンバが使用中の場合にはそれをL/Lに戻して、デッドロックの可能性を解消する
B
232
配向チャンバに配向済みウェハが無いか先読みし、行先チャンバが使用中の場合にはそれをL/Lに戻す
C
234
戻されたデッドロックウェハがロードロックチャンバに無いか先読みして、加工のスケジュールを組む
D
236
事前配向されたウェハがロードロックチャンバに無いか先読みして、加工のスケジュールを組む
E
238
ウェハ移動待ち行列のウェハの優先順位を決定する
F
240
ウェハ移動待ち行列内の全てのウェハを移動する
G
戻る

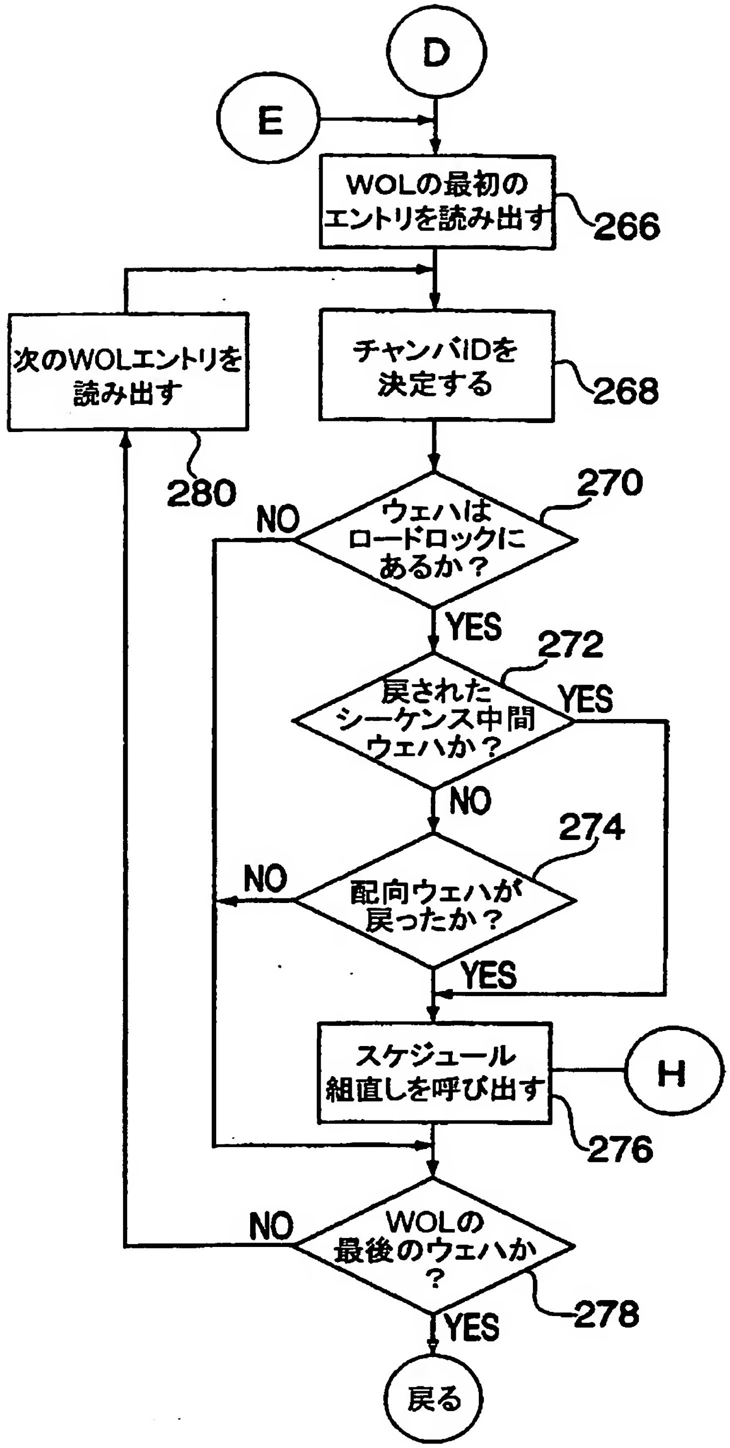
【図８ｂ】


B
242
グループチャンバか？
NO
YES
244
チャンバデータ構造のレシピ残り時間の
情報を使用して、次に可能なチャンバを選択する
チャンバID
先読みし、チャンバデータ構造の情報を
使用して、チャンバが次の移動に利用可能か
どうかを決定する
246
248
チャンバは利用可能か？
NO
YES
供給源、
行先チャンバID
デッドロックフラッグをセットし、次の
処理チャンバIDをWOLに保存する
行先チャンバ＝ロードロック
250
ウェハ移動待ち行列データ構造に
ウェハ識別データを入力する
252
戻る

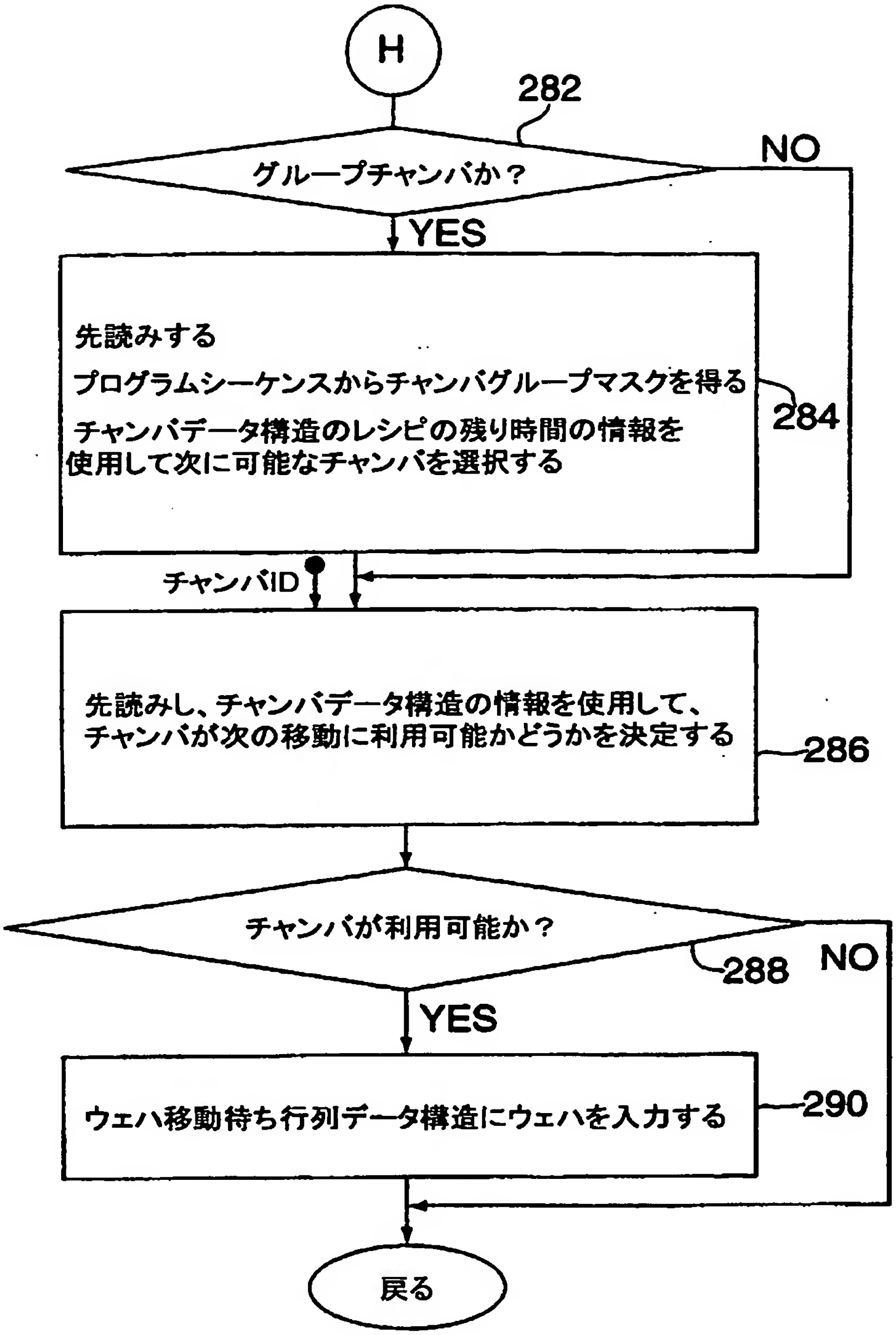
【図８ｃ】


C
254
グループチャンバか？
NO
YES
チャンバデータ構造のレシピの
残り時間の情報を使用して、次に可能な
チャンバを選択する
256
チャンバID
先読みし、チャンバデータ構造の情報を
使用して、チャンバが次の移動に利用可能か
どうかを決定する
258
260
チャンバは利用可能か？
NO
YES
供給源、
行先チャンバID
事前配向フラッグをセットし、次の
処理チャンバIDをWOLに保存する
行先チャンバ＝ロードロック
262
ウェハ移動待ち行列データ構造に
ウェハを入力する
264
戻る

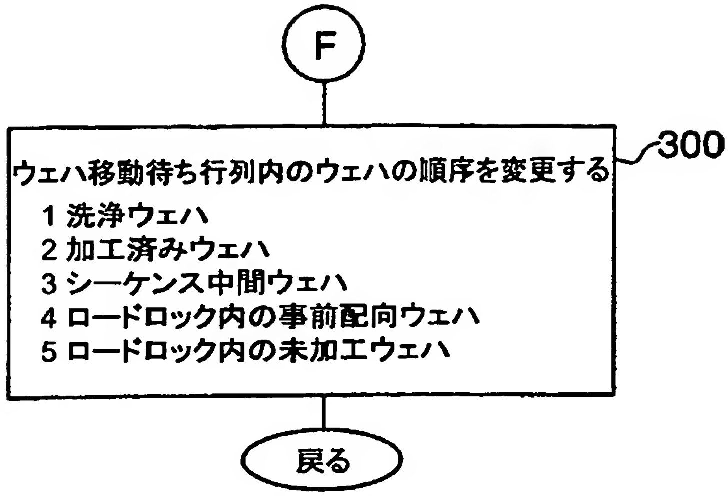
【図８ｄ】


D
E
WOLの最初の
エントリを読み出す
266
次のWOLエントリを
読み出す
280
チャンバIDを
決定する
268
270
NO
ウェハは
ロードロックに
あるか？
YES
272
戻された
シーケンス中間
ウェハか？
YES
NO
274
NO
配向ウェハが
戻ったか？
YES
スケジュール
組直しを呼び出す
H
276
NO
WOLの
最後のウェハか
？
YES
278
戻る

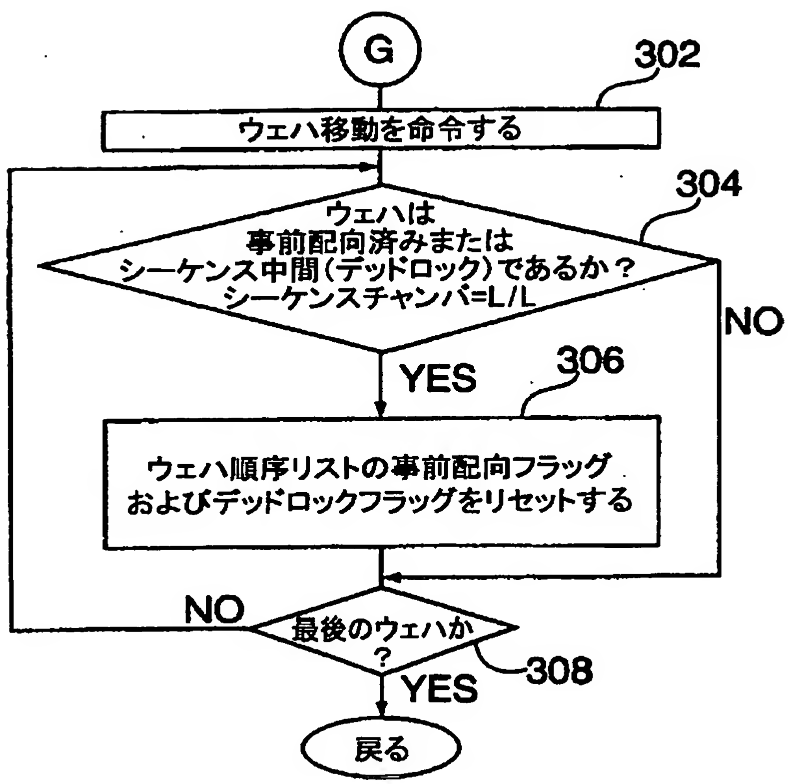
【図８ｅ】


H
282
グループチャンバか？
NO
YES
先読みする
プログラムシーケンスからチャンバグループマスクを得る
チャンバデータ構造のレシピの残り時間の情報を使用して次に可能なチャンバを選択する
284
チャンバID
先読みし、チャンバデータ構造の情報を使用して、チャンバが次の移動に利用可能かどうかを決定する
286
チャンバが利用可能か？
288
NO
YES
ウェハ移動待ち行列データ構造にウェハを入力する
290
戻る

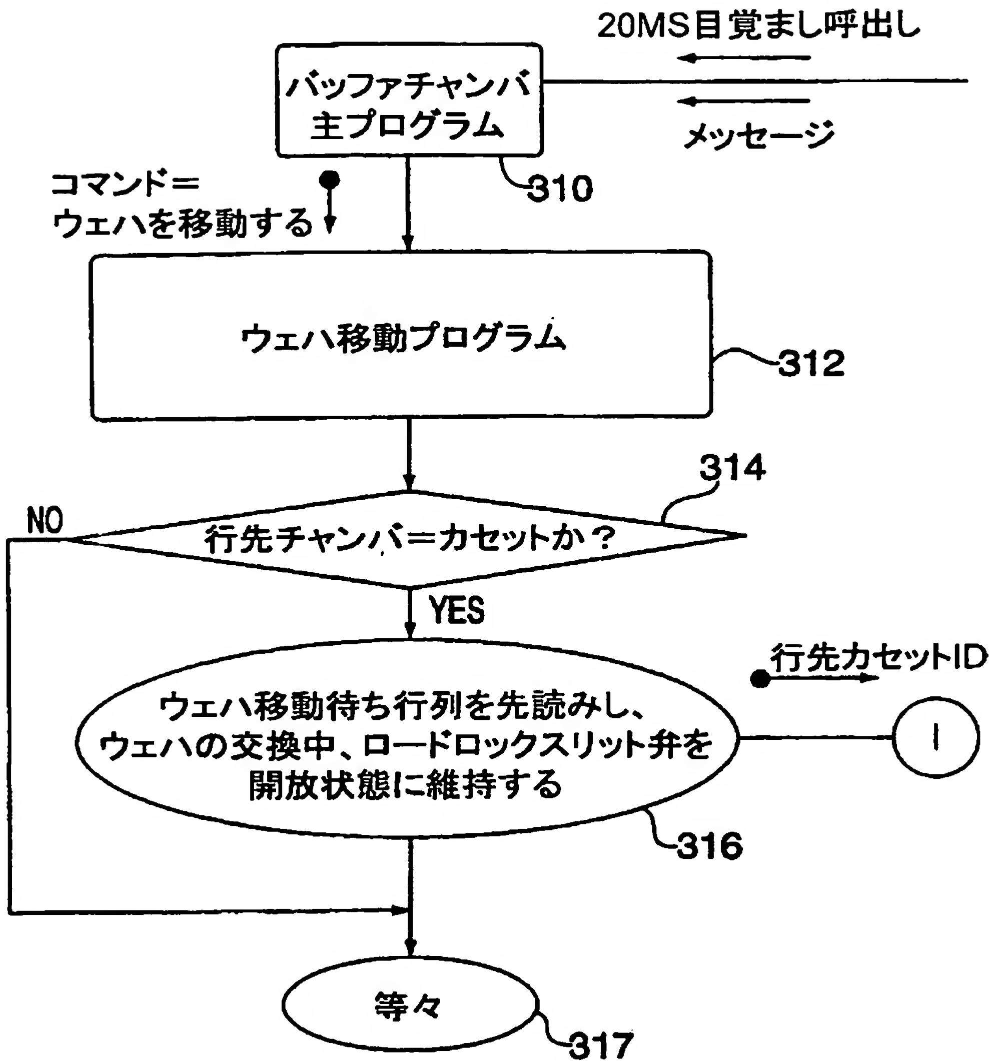
【図８ｆ】


F
ウェハ移動待ち行列内のウェハの順序を変更する
1 洗浄ウェハ
2 加工済みウェハ
3 シーケンス中間ウェハ
4 ロードロック内の事前配向ウェハ
5 ロードロック内の未加工ウェハ
300
戻る

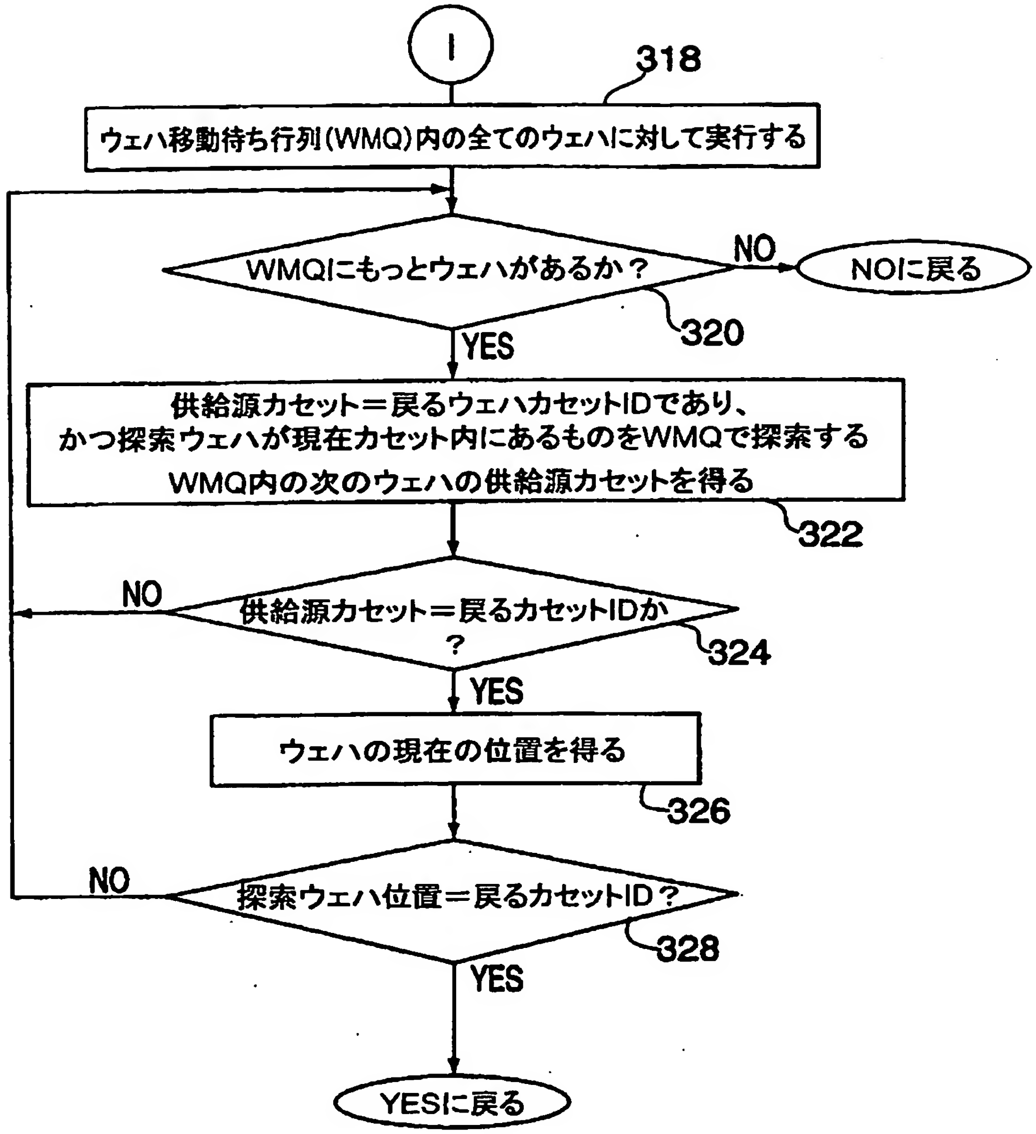
【図８ｇ】


G
302
ウェハ移動を命令する
304
ウェハは
事前配向済みまたは
シーケンス中間(デッドロック)であるか？
シーケンスチャンバ=L/L
NO
YES
306
ウェハ順序リストの事前配向フラッグ
およびデッドロックフラッグをリセットする
NO
最後のウェハか
？
YES
308
戻る

【図９ａ】


20MS目覚まし呼出し
メッセージ
バッファチャンバ主プログラム
310
コマンド＝ウェハを移動する
ウェハ移動プログラム
312
314
NO
行先チャンバ＝カセットか？
YES
行先カセットID
ウェハ移動待ち行列を先読みし、ウェハの交換中、ロードロックスリット弁を開放状態に維持する
316
等々
317

【図９ｂ】


I
318
ウェハ移動待ち行列(WMQ)内の全てのウェハに対して実行する
WMQにもっとウェハがあるか？
NO
NOに戻る
320
YES
供給源カセット＝戻るウェハカセットIDであり、
かつ探索ウェハが現在カセット内にあるものをWMQで探索する
WMQ内の次のウェハの供給源カセットを得る
322
NO
供給源カセット＝戻るカセットIDか？
324
YES
ウェハの現在の位置を得る
326
NO
探索ウェハ位置＝戻るカセットID？
328
YES
YESに戻る

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 99/03113

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6　H01L21/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6　H01L　G05B

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 105 362 A (KOTANI) 14 April 1992 see the whole document	1,13,20
A	EP 0 810 632 A (APPLIED MATERIALS) 3 December 1997 see the whole document	1,13,20
A	GB 2 296 818 A (NEC CORPORATION) 10 July 1996 see the whole document	1,13,20
A,P	EP 0 837 494 A (APPLIED MATERIALS) 22 April 1998 see the whole document	1,13,20

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
16 June 1999	24/06/1999

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Oberle, T

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US 99/03113

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5105362 A	14-04-1992	JP 1997308 C	08-12-1995
		JP 6016475 B	02-03-1994
		JP 63249328 A	17-10-1988
		US 4901242 A	13-02-1990
		US 5111404 A	05-05-1992
EP 0810632 A	03-12-1997	NONE	
GB 2296818 A	10-07-1996	CN 1131813 A,B	25-09-1996
		JP 2702469 B	21-01-1998
		JP 8236413 A	13-09-1996
		US 5745364 A	28-04-1998
EP 0837494 A	22-04-1998	JP 10189687 A	21-07-1998

フロントページの続き

(72)発明者　ラヴィ，　マイカル
アメリカ合衆国，　カリフォルニア州，
サン　ノゼ，　クラレンドン　ストリート
7136

Ｆターム(参考)　5F031 CA02 FA01 FA07 FA11 FA12
MA04 MA11 MA28 MA29 MA31
NA05 PA03 PA05